PROBLEM STRUCTURING AND ANALYSIS METHOD BASED ON SOFT SYSTEM METHODOLOGY FOR TEACHING REQUIREMENTS ENGINEERING

RAFIA NAZ MEMON

THESIS SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

2014

Abstract

The discipline of Requirements Engineering (RE) is hard for students to understand and challenging for lecturers to teach due to its complex nature. In the literature, researchers have presented a number of Requirements Engineering Education (REE) problems in universities, as well as RE problems in the industry, which can be addressed by providing proper REE at universities.

However, no effort has been made to systematically compile, and analyse these problems so that it is possible to have a bird's-eye view of REE problems. This analysis and presentation of the problems is referred to as "an integrated view of the REE problems". Some problems have been reported and investigated by researchers while some remain only as reported problems. Therefore, this research aims to come up with an integrated view of REE problems, followed by selecting one problem area to be the research focus and addressing the selected problem by formulating a method.

In the first phase of the study, an integrated view of REE problems was produced. A survey was then performed among students and lecturers who have studied and taught RE course to verify the REE problems presented in the integrated view and to identify a selected research focus. This leads to the identification of the research focus which is teaching problem structuring and analysis.

The literature survey on problem structuring and analysis showed that it has not been explicitly emphasised in the RE course but thought to be covered using requirements analysis methods. For students with no industrial experience, it is difficult to learn problem structuring and analysis using these methods. Therefore, in the second phase of the research, a light-weight method was proposed to be taught to undergraduate software engineering students in the RE course to enable them to understand and perform the problem structuring and analysis. The method is referred to as L-Soft to portray that it has transformed the idea of Soft system methodology into a light-weight

(simple structure and easy to apply procedure) RE method. In order to facilitate the understanding and application of the L-Soft method, students were provided with a step-by-step guide on performing the problem structuring and analysis, a glossary of terms used in the method and a sample of solved case studies. A web-based tool for L-Soft was developed to provide learning support to the students.

In the final phase of the study, the appropriateness of the method for performing the problem structuring and analysis was validated using feature analysis performed by lecturers and software engineers. The perceived adoption in practice was validated using method acceptance testing performed by the lecturers and software engineers and the results showed that the L-Soft method has a high likelihood of being adopted in practice. Finally, an experimental study was performed among undergraduate software engineering students to validate the method's successfulness. The results showed that L-Soft was more efficient and effective than an existing analysis method and have a higher likelihood of being adopted into practice rather than the existing analysis method.

Abstrak

Disiplin Keperluan Kejuruteraan (RE) adalah sukar bagi pelajar untuk memahami dan mencabar bagi pensyarah untuk mengajar kerana fitrahnya yang rumit. Dalam kesusasteraan saintifik, ramai penyelidik telah membincangkan beberapa masalah Kejuruteraan Keperluan Pendidikan (REE) di universiti, serta masalah RE dalam industri, yang boleh ditangani dengan menyediakan pendidikan RE yang baik di universiti.

Walau bagaimanapun, tidak ada usaha telah dibuat secara sistematik untuk mengumpul, mengkaji dan menganalisis masalah-masalah ini supaya membolehkan kita untuk mempunyai pandangan meluas tentang masalah REE serta kumpulan dan kategori masalah dan hubungan antara mereka. Analisis dan pembentangan masalah ini boleh dirujuk sebagai "pandangan bersepadu masalah REE". Sebahagian masalah telah dilaporkan dan disiasat oleh penyelidik dan sebahagian masih lagi kekal hanya sebagai masalah yang dilaporkan. Oleh itu, kajian ini bertujuan untuk menghasilkan pandangan bersepadu masalah REE, diikuti dengan pemilihan satu bidang masalah yang boleh menjadi tumpuan penyelidikan dan menangani masalah yang dipilih dengan perumusan suatu kaedah.

Dalam fasa pertama kajian, pandangan bersepadu masalah REE telah dihasilkan. Satu soalselidik telah dilakukan di kalangan pelajar dan pensyarah yang telah belajar dan mengajar kursus RE. Tujuannya adalah untuk mengesahkan masalah REE yang dibentangkan dalam pandangan bersepadu dan untuk mengenal pasti fokus kajian. Ini membawa kepada pengenalpastian fokus kajian iaitu pengajaran penstrukturan masalah dan analisis.

Kajian kesusasteraan mengenai penstrukturan masalah dan analisis menunjukkan bahawa ia tidak jelas ditekankan dalam kursus RE tetapi dianggap telah dicangkupi

menggunakan kaedah analisis keperluan. Bagi pelajar yang tidak mempunyai pengalaman industri, adalah sukar untuk melaksanakan penstrukturan masalah dan analisis menggunakan kaedah kaedah tersebut. Oleh itu, dalam fasa kedua penyelidikan, suatu kaedah ringan telah dicadangkan untuk diajar kepada pelajar-pelajar ijazah kejuruteraan perisian dalam kursus RE untuk membolehkan mereka memahami dan melaksanakan proses penstrukturan masalah dan proses analisis. Kaedah ini dipanggil *L-Soft* sebagai suatu kaedah yang telah mengubahsuai idea metodologi *Soft System* ke bentuk satu kaedah RE ringan (mudah struktur dan mudah untuk memohon prosedur). Dalam usaha untuk memudahkan pemahaman dan aplikasi kaedah *L-Soft*, pelajar disediakan dengan panduan langkah-demi-langkah untuk melaksanakan proses penstrukturan masalah dan proses analisis, glosari istilah yang digunakan dalam kaedah dan kajian kes yang telah diselesaikan. Satu peralatan perisian berasaskan sesarang bagi L-Soft telah dibangunkan, untuk menyediakan sokongan pembelajaran kepada pelajar.

Dalam fasa akhir pengajian, kesesuaian kaedah telah disahkan menggunakan analisis ciri yang dijalankan oleh pensyarah dan jurutera perisian. ‘*Perceived adoption*’ dalam amalan telah disahkan dengan menggunakan pengujian penerimaan pakar oleh pensyarah dan jurutera perisian. Keputusan menunjukkan kaedah L-Soft mempunyai kecenderungan tinggi untuk di gunapakai dalam amalan. Akhirnya, satu kajian eksperimen telah dijalankan di kalangan pelajar sarjana muda kejuruteraan perisian untuk mengesahkan keupayaan kaedah yang dicadangkan. Keputusan menunjukkan L-Soft lebih cekap dan berkesan dari satu kaedah sedia ada dan mempunyai kecenderungan yang lebih untuk digunapakai dalam amalan dibandingkan dengan satu kaedah sedia ada.

ACKNOWLEDGEMENTS

In the name of Allah, Most Gracious and Most Merciful.

The dissertation process for the PhD degree is considered as a long, lonely, challenging, yet rewarding path to pursue. For me a lot of reading, thinking, writing, discussion, frustration, and joy came along with the process. I am grateful for all the bounties that Allah has showered on me which enabled me to complete this thesis.

My sincere gratitude goes to many people who have given me unconditional support, love, encouragement, and care throughout this process and who were fundamental for helping me to finish the thesis. In particular, Assoc. Prof. Dr. Rodina Ahmad and Prof. Dr. Siti Salwah Salim, both of my supervisors who have each contributed great thoughts and perspectives and provided guidance throughout the various revisions of this thesis. This thesis could not have been written without their supervision, guidance and support.

I would like to thank Quaid-e-Awam university of Engineering, Science & Technology, Nawabshah, Pakistan for providing me with the financial and administrative support that is most valuable for international doctoral students. All this support has allowed me to concentrate my efforts on my research.

Thank you also to University of Malaya and to my faculty, Faculty of Computer Science and Information Technology for giving me this opportunity. I would like to acknowledge all the wonderful software engineering students who have participated in the studies conducted during this research for their assistance and cooperation. I gratefully acknowledge the lecturers and software engineers who took part in the surveys.

Lastly, and most important of all I would like to thank my parents who each means a world to me, for all their prayers and support. During the whole period, I could always count on their support, and most importantly, their belief in me. It is to them that I dedicate this work.

Table of contents

List of figures

List of tables

Chapter 1 - Introduction

1.1 Background and motivation

Requirements Engineering (RE) is concerned with the field of Software Engineering. It is the most important albeit difficult phase of software development life cycle. RE is about capturing the requirements of customers and analysing, modelling and validating those requirements and presenting them in a software requirements specification, which is the final output of RE. The software is then developed based on these requirements. Failure to identify these requirements accurately will result in failure to meet the project goals and satisfy the customers.

Research into software development has found that the major cause of failures and deficiencies in software projects stems from the poor fulfilment of RE activities by software engineers. The most common reason for this is the lack of required knowledge and skills of software engineers working on these projects (Armarego & Minor, 2005). Due to the lack of RE in most academic programs, software developers have to learn the RE activities during the job (Jiang, Eberlein, & Far, 2005). This leaves them with the lack of RE skills and knowledge. Therefore, it is important to provide sufficient Requirements Engineering Education (REE) to students at university level before they become software engineers and part of the workforce (Berenbach, 2005). Unfortunately in most computer science or software engineering programs, RE is not a highly visible topic due to a number of reasons, such as the complex and theoretical nature of RE that makes it difficult for students to understand. Students also do not perceive it as interesting or glamorous and find it difficult to define research topics in RE.

Lecturers too find challenging to teach, specifically in finding the best way to prepare students for the RE activities in a limited time. Moreover, a requirements engineer's job is not a glamorous job like a developer's or an architect's; these jobs are sometimes considered as dead end and boring. In spite of all these feedbacks, RE is a vital part of software development life cycle (Berenbach, 2005).

Despite its importance, there is not a lot of published work on the teaching of RE (Callele & Makaroff, 2006). However in the last few years, there has been an increasing emphasis on university curriculum for RE for undergraduate as well as postgraduate students. The most important challenge in REE is to prepare students for learning sufficient skills to perform RE activities within the limited time and resources available at learning institutions (Yusop, Mehboob, & Zowghi, 2007). In order to meet that challenge, it is urgent to identify an enhanced pedagogical approach to incorporate a learner-centred design in the development of curriculum and instructional strategies, to develop a general and flexible curriculum framework along with the supporting materials, and to exploit new technologies for on-campus learning (Adroin, 2000).

The REE currently provided to students still incorporates the traditional methods of teaching basic concepts of processes, models and methodologies. However through typical lectures, students do not learn the skills that the industry requires (Beatty & Agouridas, 2007). REE should be aimed at achieving the industrial relevance so that students will be able to cope with large scale software development projects, and the challenges and proven techniques related to industrial development of software (Wohlin & Regnell). A typical industrial project has two to four thousand requirements, whereas a RE project in academic programs is usually a very simple project in order to keep the material manageable and not to overwhelm the students. However, students should learn to manage scale before they are placed in difficult situation i.e. the industry (Berenbach, 2005).

(Regev, Gause, & Wegmann, 2009) suggested that for teaching RE, a curriculum is required in which the students do internships in the middle of studies in order to get the experiences required to fully appreciate and understand the RE practices. (Beatty & Agouridas, 2007) suggested that there is a need to teach students how to use justifiable assumptions where the information is incomplete and how to integrate the required knowledge and skills. Students should understand the effect of poor requirements on projects that can lead them to an in depth understanding of RE. Also, learning through doing is more effective than learning through being told. Students should be taught to work on different problematic situations similar to those that they will encounter in the industry (Beatty & Agouridas, 2007).

1.2 Problem statement

RE is the most difficult stage of software development for students to learn and for lecturers to teach (Gibson, 2000). Despite being a relatively new field in Software Engineering, it has attracted the attention of both academics and practitioners because it is one of the most important, difficult and problematic phase of software development. The education industry nowadays is working on producing more people who can perform the role of a Requirements Engineer but the gap between what the industry needs and what the graduates learn from the RE courses is still very vast (Scheinholtz, 2007). A significant number of studies have been done in the area of REE in which researchers have reported a number of REE problems in universities, as well as RE problems in industries that can be addressed by providing REE in universities. However, at the time of this study, no efforts have yet been made to systematically compile, review and analyse these problems and to present them in such a way that it would be possible to see a collection of REE problems together. Therefore, as one of the steps towards bridging the gap between REE provided to students and RE practices in industry, this research identifies and analyses REE problems in an integrated representation that can enable one to see together the whole range of problems in REE with an intent of bringing to the attention of a bigger audience the problem areas in REE. We have called the analysis and presentation of the entire range of REE problems as "an integrated view of the problems."

Of all the REE problems reported in several studies in the literature, only a few problems are well-investigated while others are either only reported or less-investigated by researchers in the literature. Most of the studies in RE examine the importance perceived by stakeholders; less importance have been given to the activities which are necessary to perform RE in detail, and also whether the skills and knowledge for these activities are included in the curricula or not (Minor & Armarego, 2005). Therefore, there is a need to further understand and clarify REE problems with the objective of formulating an enhanced approach for learning and teaching RE. Also, more practical approaches should be employed and ventured to produce better results from the course.

1.3 Research objectives

The fundamental goal of this research effort is to understand and analyse REE problems presented in the literature as well as reported by RE lecturers and students, and to propose a light-weight method understandable by students in order to enhance their skills in learning and applying RE activities. In order to meet this goal, the overall objectives of the research are:

1) To identify and analyse REE problems from the literature and present them in an integrated representation and to verify REE problems from students and lecturers.
2) To address the one selected problem by formulating a method suitable to be taught to undergraduate software engineering students in RE course.
3) To design and develop a software prototype as a learning support tool to support the proposed method.

4) To validate the proposed method.

1.4 Research Methodology

This research uses constructive methodology which is a widely used research method in software engineering and computer science. The key idea of constructive research is the construction, based on the existing knowledge used in novel ways, with possibly adding a few missing links (Crnkovic, 2010). Constructive research builds on both prior theory and the practical relevance of the work (El-Shamy, 2001). It was chosen because it offers a method for producing novel solutions to address practically and theoretically relevant problems (Smith, 2009). The research methodology comprises of five main phases. The activities in each phase are summarized below:

1.4.1 Phase 1: Find practically relevant problem

A literature survey was performed in order to identify REE problems. Topics covered in the literature review include REE problems in universities as well as RE problems in industries that can be addressed by providing proper REE in universities. An integrated representation presenting all the identified REE problems was developed that presents the problems along with the analysis performed on those problems. Chapter 2 and 3 cover the activities of phase 1.

1.4.2 Phase 2: Obtain an understanding of the topic and problem

In order to verify REE problems presented in the integrated view and to formulate the research focus that contributes in addressing one of the main REE problems, investigations were carried out among students and lecturers. These investigations confirm almost all problems presented in the integrated view. Through further analysis of REE problems, one practical problem was selected as the research focus for this study.

The selected problem was then investigated from the point of view of lecturers in order to gather their suggestions and recommendations. Chapter 4 covers the activities of phase 2.

The findings from Phase 1 and Phase 2 were intended to fulfil objective 1.

1.4.3 Phase 3: Innovate: Construct a solution idea

Another literature survey was performed in order to study the state-of-the-art of selected research focus and the related method proposed. As a result, a light weight method based on the idea of soft system methodology was formulated to be taught to undergraduate software engineering students in RE course. Students were provided with the guidelines and templates to help them understand and apply the method. Chapter 5 and 6 cover the activities of phase 3. Activities performed during this phase were intended to fulfil objective 2.

1.4.4 Phase 4: Develop a prototype

A software prototype of the proposed method was developed as a learning support tool to be used by students (tool users) and lecturers (tool admin). It is a web-based tool that automates the steps of the proposed method. Using the tool, a lecturer can apply the method to solve example problems, provide solutions to guide students, add new problems as an input to students to solve, and check the problems solved by students. As for students, the tool can act as a learning support tool. They are provided with the 'Getting Started' demo, solved case studies and guidelines that can help them perform the process. The automated application of the steps of the method makes it easier for them to apply the process. Chapter 7 covers the activities of phase 4. This phase was performed in order to fulfil objective 3.

1.4.5 Phase 5: validate the solution

In the final phase of the study, three types of validations were performed to validate the method. They are:

i) Feature analysis: The qualitative validation performed to assess the method's appropriateness by rating the features that the method aimed at supporting and the RE process that it should possess. Feature analysis is performed by lecturers and software engineers who will assess and rate the features after studying and applying the method (Qualitative survey).

ii) Method acceptance testing: Lecturers and software engineers studied and applied the method and then filled the post-task questionnaire for method acceptance.

iii) Formal experiment: The successfulness of the method in terms efficiency, effectiveness and perceived adoption in practice was validated through a formal experiment performed among undergraduate software engineering students.

Chapter 8 covers the activities of phase 5. These activities were intended to fulfil objective 4.

Figure 1.1 shows the phases involved in this study, the activities performed in each phase and their mapping with the research objectives.

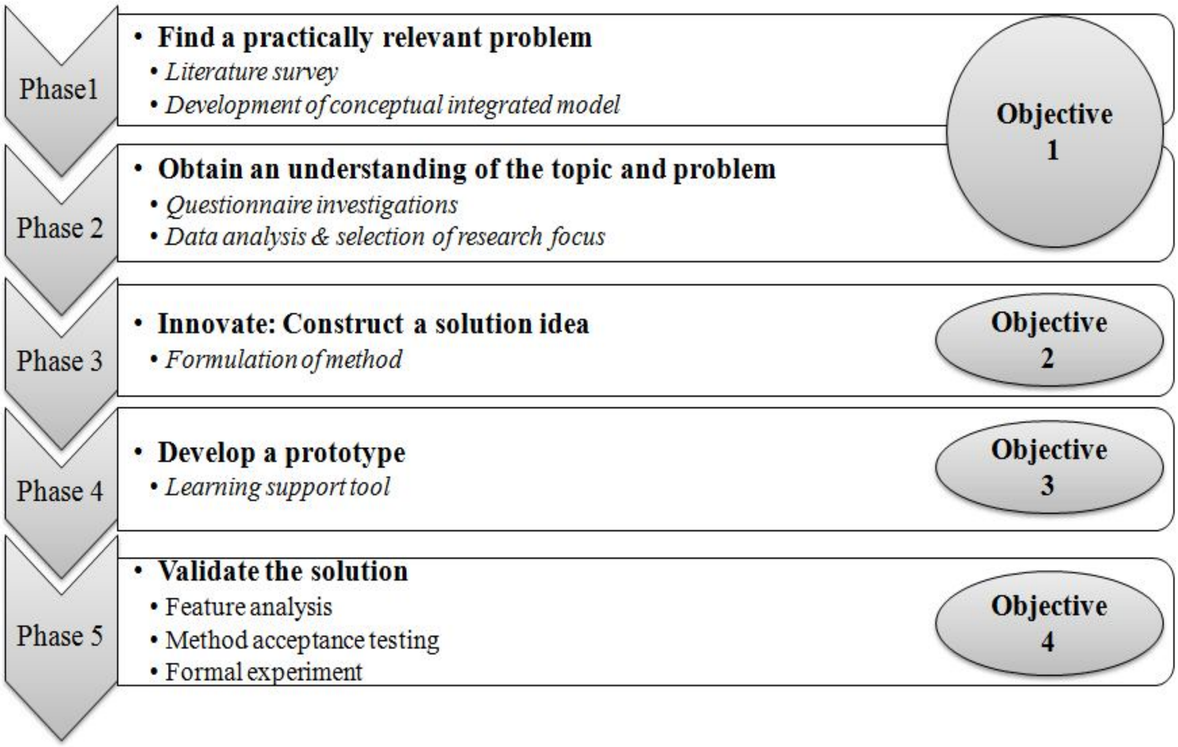


Figure 1.1: The relationship of the research methodology phases with the research objectives.

This study constitutes two major parts. The first part reviewed the literature on issues in REE, and identified and analysed in detail REE problems and presented them in an integrated representation. A survey was then performed among students and lecturers who have studied and taught RE course to verify the REE problems presented in the integrated view. Further analysis of the REE problems presented in the literature and the investigations results resulted in the selection of the research focus to be addressed in the second part of the research. In this part of the research, a light weight RE method was proposed to be taught in RE course in universities in order to address the selected REE problem and a prototype was developed as a learning support tool. The method was then validated by the students, lecturers and software engineers.

1.5 Research contributions

The goal of this dissertation is to support REE by attracting RE researchers to the REE problems and by offering a light-weight method to be taught in RE course in universities. Thus, the contributions of this research are as follow: -

a. From the literature, REE problems are classified and presented in an integrated representation (called integrated view). The **integrated view** is produced to help RE researchers to see a whole range of REE problems at once.

b. The **development of a light-weight method** based on the idea of soft system methodology to help students to quickly learn and apply problem structuring and analysis in RE.

c. The **development of a software prototype** that automates the steps of the proposed method to support the teaching of the method for lecturers and the learning of the method for students.

1.6 Significance of the research

The integrated representation of REE problems produced in this research can attract RE researchers towards the REE problems as it presents the problems together with their analysis information. This enables researchers to know about the REE problems that are already addressed as well as those problems that still need to be addressed. Thus, the integrated view can serve as a background for further REE research.

In addition, as part of this research effort, a light-weight method was proposed as a way to introduce an important aspect of RE to students in RE course. Students are assisted in understanding and applying the method by providing them with guidelines, templates, solved case studies and a prototype tool. As a result of this research, RE lecturers and students will have a new method available to introduce an important RE aspect, thus advancing REE and also provide requirements engineers with a better understanding of RE process which will also benefit industry.

The central argument of this dissertation is that light-weight RE methods should be taught to students because these methods often allow easy adaptation and introduction of new concepts. The objective is to introduce RE process in an easy and understandable manner. Research into light-weight and easy-to-adopt RE methods will benefit both academia and industry. In academia, if students understand how to apply RE process, their transition to industry may be smoother. In industry, if requirements engineers understand and practice good RE, the result will likely lead to quality software projects that meet the needs and expectations of stakeholders.

1.7 Thesis outline

There are 9 chapters and six appendices in this thesis. The overall structure of the thesis is illustrated in figure 1.2.

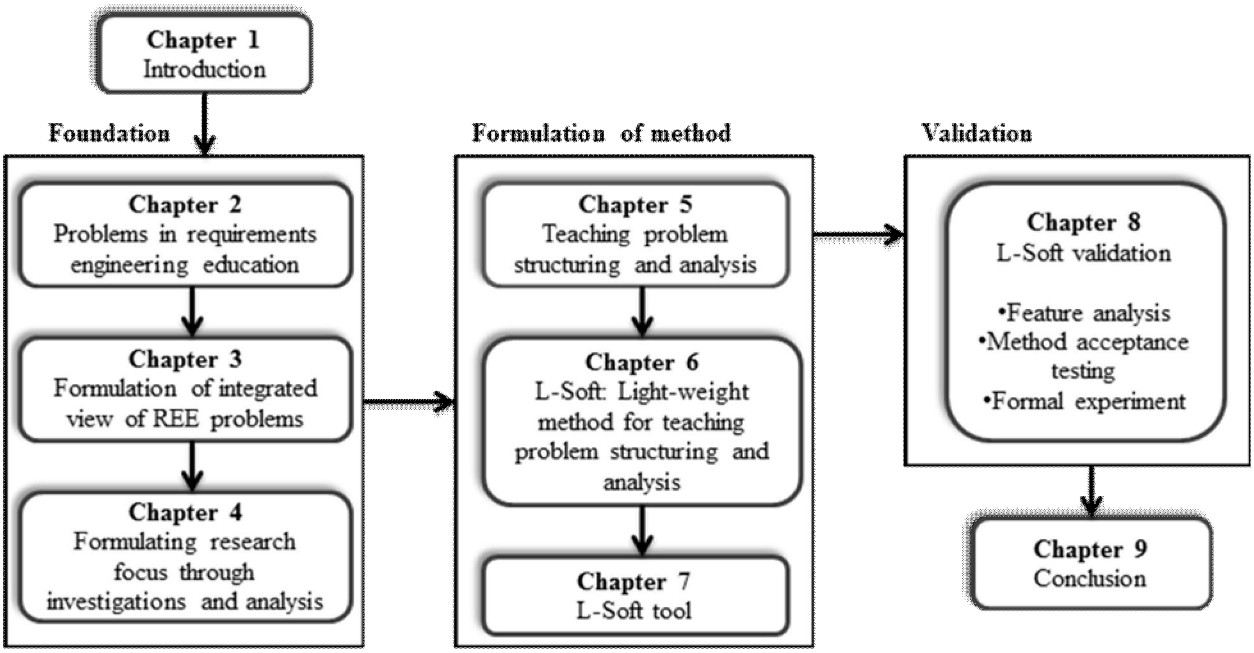


Figure 1.2: Structure of the thesis

Chapter 1- Introduction- Introduces the research topic and provides an overview of the dissertation including research problem, research objectives, and research methodology and contribution. It also presents the structure of the thesis.

Chapter 2- Problems in requirements engineering education- Gives an introduction of requirements engineering (RE) and requirements engineering education (REE) and discusses in general, the requirements engineering education provided to students in different institutions. It then discusses researchers' work on REE and presents REE problems reported by researchers in the literature.

Chapter 3- Formulation of an integrated view of REE problems- Presents the analysis performed on the REE problems to develop a conceptual integrated view that presents all REE problems together with their related information.

Chapter 4- Formulating research focus through investigations and analysis- Presents the result of investigations performed on students and lecturers in order to verify the problems presented in the integrated view and to formulate the research focus. Research gaps in REE were identified and through further analysis, a practical problem was selected as the focus for this research that needs to be addressed further. The selected problem is "teaching problem structuring and analysis".

Chapter 5- Teaching problem structuring and analysis- Gives an introduction to problem structuring and analysis in RE and presents related work in this area from the literature. It then presents the research gap that needs to be filled in this research.

Chapter 6- Light-weight method for teaching problem structuring and analysis (L-Soft) - Presents the proposed light-weight method in detail.

Chapter 7- L-Soft tool- Presents the structure and working of a web-based tool for the proposed method.

Chapter 8- L-Soft validation- Presents the validations performed to validate the method for performing problem structuring and analysis process and teaching it to students in RE course.

Chapter 9- Conclusion- Gives a review and conclusion of the work. Future work is suggested.

Appendix A- Student's questionnaire- Contains the questionnaire aimed at investigating and verifying the problems presented in the integrated view from students' perspectives.

Appendix B- Lecturer's questionnaire-I- Contains the questionnaire aimed at investigating and verifying the problems presented in the integrated view from lecturers' perspectives.

Appendix C- Lecturer's questionnaire-II- Contains the questionnaire aimed at verifying the selected research focus from lecturers.

Appendix D- Lecturer's and Software engineer's post-task questionnaire- Contains the post-task questionnaire aimed at lecturers and software engineers to validate the proposed method.

Appendix E- Student's post-task questionnaire on L-Soft- Contains the post-task questionnaire aimed at students to validate the proposed method.

Appendix F- Student's post-task questionnaire on existing analysis method- Contains the post-task questionnaire aimed at students to perform the comparison validation study.

Chapter 2- Problems in Requirements Engineering Education

2.1 Introduction

This chapter gives an overview of Requirements Engineering (RE) and focuses on Requirements Engineering Education (REE) including background studies on REE, available course on RE, and recommended RE course contents by ACM and IEEE-computer Society Educational Board. It then identifies the studies within the literature that have presented problems in REE, and extracts and presents the problems reported in those studies.

2.2 Requirements Engineering

RE is the first phase of development lifecycle, and all the subsequent phases are dependent on this phase. It shows the importance of this phase. This is the phase which involves gathering information about the customer's needs and defining, in the clearest possible terms, the problem that the product is expected to solve (Melonfire, 2007). The requirements process studies the work in order to devise the best possible product to help with that work. As an outcome of this process, the requirements specification is a complete description of the functionality and the behaviour of the product (Robertson & Robertson, 2012). This section presents the definitions and main activities of the RE phase.

2.2.1 Definition

RE is an inherently complex discipline and has been broadly recognized as critical to the development project success (Beatty & Agouridas, 2007). A requirement is defined as “a property that must be exhibited in order to solve some real-world problem”, whereas the term Requirements Engineering refers to the “systematic handling of requirements” (Abran, Moore, Bourque, Dupuis, & Tripp, 2004).

The final output of RE is the software requirements specification (SRS), which is defined as: “A document that clearly and precisely describes each of the essential requirements (functions, performance, design constraints, and quality attributes) of the software and external interfaces. Each requirement is defined in such a way that its achievement can be objectively verified by a prescribed method, for example, inspection, demonstration, analysis, or test.” (Thayer, Bailin, & Dorfman, 1997)

Having defined requirements and SRS, the following three definitions of RE are given. According to (Wahono, 2003), RE is “A systematic approach to eliciting, organizing, and documenting the requirements of the system, and a process that establishes and maintains agreement between the customer and the project team on the changing requirements of the system”. This definition gives a very intuitive description of RE and is highly related to representational, social, and cognitive issues. On the other hand, a more comprehensive definition of RE is given by (Zave, 1997): “RE is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behaviour, and to their evolution over time and across software families.” Whereas according to (Wieringa & Ebert, 2004), RE is a multidisciplinary field that includes the characteristics of Software Engineering, System Engineering, product management and the psychology in it. They define RE as “the branch of systems engineering concerned with the desired properties and constraints of software-intensive systems, the goals to be achieved in the software’s environment, and assumptions about the environment”. From these definitions, it reflects that RE is multidimensional discipline in nature which not only relates to technical issues and problems, but also closely relates to managerial, organizational, economic, and social issues and problems. It might not only be a front end process, but might also be a part of the later stages of software engineering projects.

2.2.2 Requirements engineering activities

The RE phase consists of the following five core activities.

2.2.2.1 Requirements Elicitation and Analysis

Requirements elicitation (also called requirements acquisition, requirements capture, requirements discovery, requirements gathering) is a process of "identifying needs and bridging the disparities among the involved communities for the purpose of defining and distilling requirements to meet the constraints of these communities" (SEI, 1991). While requirements analysis is a process of understanding and reviewing requirements. The most difficult part of this phase is to ensure effective communication between various stakeholders and the elicitation of tacit knowledge (Jiang, 2005). The output of the requirement elicitation and analysis phase is a collection of elicitation notes or an informal document which describes the elicited requirements (Bray, 2002). The requirements in this phase are still largely unprocessed and may contain many irrelevancies, inconsistencies and omissions (Jiang, 2005). A lot of techniques can be used in the requirements elicitation and analysis process such as interviews, ethnography, contextual query, facilitated meetings and observations. (Bray, 2002).

2.2.2.2 Requirements Modelling

During requirements modelling, requirements are modelled to resolve possible conflicts by negotiation between stakeholders (Jiang, 2005). The elicitation and analysis process provides the input to this process. The output of the process is a consistent and complete set of requirements. Some typical techniques that can be used during this phase are Unified Modelling Language (UML), Specification and Description Language (SDL), Structured Analysis Structured Design (SASD), Goal-based techniques, and Petri Nets.

Requirements can be modelled using three different kinds of languages or notations, informal language, semi-formal language and formal language (Jiang, 2005).

2.2.2.3 Requirements Documentation

Requirements documentation is the process of documenting the agreed requirements at an appropriate level of detail in the most suitable notation based on a well-defined document structure. This process has a very close relationship with requirements management (Jiang, 2005). The documentation process receives its input from the analysis and negotiation process. The output of the process is a well-structured and defined specification, which, however, still needs to be verified and validated (Jiang, 2005). As requirements can be modelled in informal (natural language), semi-formal (diagrams, graph) and formal languages (Mathematical notations), the requirements documentation may contain models in any of the three different notations or combinations of them. Even if the system is documented in a formal notation, an informal document is usually also required to improve understandability of the SRS (Jiang, 2005).

2.2.2.4 Requirements Verification and Validation

Requirements verification and validation (V&V) is the process of examining the requirements document to ensure that it is unambiguous, consistent and complete, and that the stakeholders are satisfied with the final requirements specification (Jiang, 2005). The task of verification is to check whether the requirements comply with given constraints, and are consistent, complete and unambiguous. This is usually done by formal verifications or inspections. The task of validation is to certify that the specified requirements comply with the given user and customer intentions. This means that the requirements need to be expressed in a notation that is understandable by the customer (Eberlein, 1997).

The output of the requirements documentation process is the input of the verification and validation process. The output of the V&V process is the finalized requirements specification document agreed and authorized by all stakeholders. The techniques used most often for this process are Formal requirements inspection, Requirements testing and Requirements checklist (Jiang, 2005).

2.2.2.5 Requirements Management

Requirements management is the process of identifying, organizing, documenting and tracking changing requirements in a project as well as the impact of these changes. It is an on-going task throughout the whole RE process and might span the whole software lifecycle (Jiang, 2005). During requirements engineering, system development and operation, new requirements are discovered and current requirements are changed. This evolution of requirements has to be managed in order to ensure high-quality specifications. The management includes issues such as information storage, organization, traceability, analysis, visualization and documentation. Additionally, relationships between requirements, as well as dependencies between requirements documents, have to be recorded. Although requirements management may look like an overhead in the beginning, it is usually rewarded by better customer satisfaction and lower overall system development costs (Eberlein, 1997).

Figure 2.1 summarizes the information about each RE activity. The rectangular boxes presents RE activities, the input and output of each activity have been shown through labelled arrows. The output of every activity is the input of the next activity. The most common techniques of each activity have been shown by the ovals connected to that activity. As Requirements Management is the activity that covers the overall RE process, therefore it has been shown by the large rectangle at the right covering all other activities.

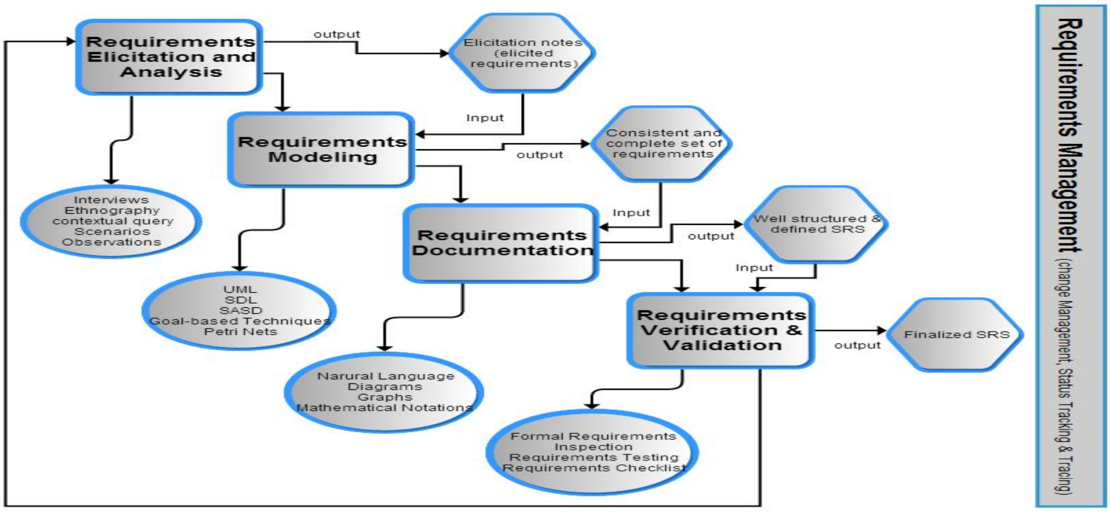


Figure 2.1: Requirements Engineering Activities

2.3 Requirements Engineering Education

This section presents background studies on REE in different contexts. First, REE will be defined, and then RE courses offered in the Computer Science/Information Technology/Software Engineering (CS/IT/SE) curriculum will be discussed. Finally, recommendations from the Association for Computing Machinery (ACM) and Institute of Electrical and Electronic Engineers (IEEE) regarding RE course content and teaching strategies will be presented.

2.3.1 Definition

According to (Rosca, 2000), the objective of a RE course is to introduce students to the process of RE and to the methods and tools available for eliciting, analysing, specifying, validating and managing requirements as well as quality criteria in the context of validating and testing the requirements. They also reported that the goal of such an education is to show the students that there is no “silver bullet” method, but instead a whole spectrum of methods must be applied in order to capture the requirements of the project according to the specifics of the system.

According to (Svahnberg & Gorschek, 2005), REE should rest on three pillars: the basic skills (such as those provided by instructional material); the state of practice (such as how RE is performed in the industry and what the industry's further needs are) and the state of the art (the current research that is being carried out in the field of REE). Whereas according to the UTS information Technology Handbook (UTS), REE should introduce students to the foundations of RE which is among the most important contributors for developing good quality software that meets the real needs of users, the front-end activities of software development and methods, and the techniques and tools that assist in the important collection of activities that makes up the requirements engineering process (Al-Ani & Yusop, 2004).

Therefore, it can be concluded that the purpose of REE is to teach students the basic concepts and skills they need to perform RE, as well as enabling them to practise performing RE activities while working on real projects.

2.3.2 RE in the CS/IT/SE curriculum

RE is being offered to students in different ways in universities, depending on the university's programme structure. The most common method involves offering RE to undergraduate students as a core subject or as an elective subject, or as a common topic in SE curriculum, or may be embedded in other related courses or in a final year project as part of a CS/IT/SE programme. In order to observe the current RE course offering, a random sample of prospectuses from a few universities in the UK, the USA, Australia, Malaysia and Canada has been examined. These examples are presented in Table 2.1.

Table 2.1: RE courses at various universities

No.	Country	University	Programme	Core	Elective	Topics
01	UK	City University London	BSc SE [a]		√	
		University College London	BSc CS [b]			√
		Manchester University	BSc CS			√
		University of Birmingham	BSc CS			√
02	USA	Samford University	BSc CS			√
		Florida Institute of Technology	BSc SE	√		
		Milwaukee School of Engineering	BSc SE	√		
		Columbus State University	BSc CS			√
		Carnegie Mellon University	BSc CS			√
03	Australia	Curtin University of Technology	BSc IT [c]			√
			BSc SE	√		
		University of New South Wales	BSc SE	√		
		University of Wollongong	BSc SE		√	
		University of Newcastle	BSc SE	√		
04	Malaysia	University of Malaya	BSc SE	√		
		University of Putra Malaysia	BSc SE	√		
		The National University of Malaysia	BSc IT			√
05	Canada	University of Toronto	BSc SE	√		
		University of Waterloo	BSc SE	√		
		University of Ontario Institute of Technology	BSc SE	√		
06	Pakistan	Mehran University of Engineering & Technology	BSc SE	√		
		University of Sindh	BSc SE	√		

[a] Bachelor of Science in Software Engineering
[b] Bachelor of Science in Computer Science
[c] Bachelor of Science in Information Technology

From these examples, it can be seen that in most BSc SE programmes, RE is offered as a core module, while some of these programmes also include RE as an elective module so that the students are equipped with a set of RE concepts (e.g., elicitation, analysis, modelling, documentation, verification, conflict resolution, team communication, problem identification), RE tools (e.g., IBM Rational RequisitePro, the Organisation Modelling Environment (OME)) and RE techniques (e.g., dealing with incomplete requirements provided by the customer, changes to the customer's requirements, involving the customer in each phase of the project). However, in most BSc CS and BSc IT programmes, RE is taught as a topic as part of a course which only exposes students to a few fundamental RE concepts and activities.

2.3.3 Recommended RE courses and teaching strategies from the ACM and IEEE Education Board

Traditionally, RE educators have developed courses based on the required textbooks, as well as academic and/or trade publications, and have then delivered these courses to students primarily in traditional classroom environments as part of CS/IT/SE degree programmes (Davis, Hickey, & Chamillard, 2005).

In order to help educators and universities to design and deliver a suitable curriculum, the ACM Education Board and the IEEE Computer Society Educational Activities Board have recommended core SE topics and guidelines for delivery in a report entitled "Software Engineering 2004 – Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering" (LeBlanc, Sobel, Diaz-Herrera, & Hilburn, 2006).

2.3.3.1 Recommended model curriculum

The section entitled "Software Modelling and Analysis" of the aforementioned report (LeBlanc et al., 2006) includes the complete core RE curriculum that is presented in Table 2.2. This recommended curriculum consists of four units, and each unit consists of essential, desirable and optional topics. The essential topics are the parts of the core curriculum that professional SE teachers agree are necessary for anyone wishing to obtain an undergraduate degree in this field. These sections are denoted with the letter E. Desirable topics are not core subjects, but should be included in the core programmes if possible; otherwise they should be considered as elective topics. These topics are denoted with the letter D. Optional topics should be considered as elective only, and are denoted with the letter O.

This recommended curriculum was specially designed to support the development of undergraduate RE courses, and can be considered as a basis to design RE programmes for universities offering RE courses. Depending on their programme structure and the available resources, universities can include advanced topics in addition to this core curriculum.

Table 2.2: Core curriculum recommended by the ACM and IEEE

Units	**Topics**	E, D, O
1. Requirements fundamentals	1. Definition of requirements (e.g., product, project, constraints, system boundaries, external, internal, etc.)	E
	2. Requirements process	E
	3. Layers/levels of requirements (e.g., needs, goals, user requirements, system requirements, software requirements, etc.)	E
	4. Requirements characteristics (e.g., testable, unambiguous, consistent, correct, traceable, priority, etc.)	E
	5. Managing changing requirements	E
	6. Requirements management (e.g., consistency management, release planning, reuse, etc.)	E
	7. Interaction between requirements and architecture	E
	8. Relationship of requirements to SE, human-centred design, etc.	D
	9. Wicked problems (e.g., ill-structured problems, problems with many solutions, etc.)	D
	10. Commercial Off The Shelf COTS as a constraint	D
2. Eliciting requirements	1. Elicitation sources (e.g., stakeholders, domain experts, operational and organisational environments, etc.)	E
	2. Elicitation techniques (e.g., interviews, questionnaires, surveys, prototypes, use cases, observation, participatory techniques, etc.)	E
	3. Advanced techniques (e.g., ethnography, knowledge elicitation, etc.)	O
3. Requirements specification & documentation	1. Requirements documentation basics (e.g., types, audience, structure, quality, attributes, standards, etc.)	E
	2. Software requirements specification	E
	3. Specification languages (e.g., structured English, UML, formal languages such as Z, VDM, SCR, RSML, etc.)	E

Table 2.2, continued

4. Requirements validation	1. Reviews and inspection	E
	2. Prototyping to validate requirements (summative prototyping)	E
	3. Acceptance test design	E
	4. Validating product quality attributes	E
	5. Formal requirements analysis	D

Overall, 75% of the recommended curriculum consists of essential topics, including RE definitions, RE processes, basic RE concepts, requirements elicitation techniques, requirements analysis and modelling techniques, requirements documentation techniques, requirements verification and validation techniques, and techniques for dealing with RE challenges in industry. A further 17% of the recommended curriculum consists of desirable topics; these topics are related to the relationship between RE and SE, RE problem analysis and structuring, RE constraints. These are supporting RE topics that, if combined with the essential topics, can provide in-depth RE knowledge to students. The overall percentage of essential and desirable topics in the core curriculum indicates that they are very important, and that one of the reasons for the problems facing educators and students in teaching and studying RE could be the lack of these topics in the curriculum. Advanced RE techniques are included as optional topics, which constitute 8% of the model curriculum. Depending on their programme structure and available resources, universities can include advanced topics in addition to the core curriculum.

2.3.3.2 Recommended RE teaching strategies

Alongside the core curriculum, guidelines for curriculum delivery are also provided in the same report (LeBlanc et al., 2006). This report states that the most common approach to teaching SE material is the use of lectures, supported by laboratory sessions or tutorials.

The dominant delivery method in most higher education institutions today is classroom-type instruction, in which the instructor presents material to a class using lectures or lecture/discussion presentation techniques, which may be augmented by appropriate laboratory work. However, the report (LeBlanc et al., 2006) recommends that SE education in the 21st century needs to move beyond the lecture format, as alternative approaches can help students learn more effectively. Therefore, we should consider a variety of approaches to teaching and learning other than those which are currently in use. Some of the recommended strategies that might be considered for supplementing, or in certain cases, even largely replacing the lecture format include problem-based learning (teaching students to solve customers' implicit and explicit problems through practice and examples), just-in-time learning (teaching fundamental material immediately before teaching the application of the same material), learning by failure (students are given a task that they will have difficulty with and are then taught methods that will enable them to carry out the task more easily in the future) and self-study materials (students work through problems in their own time, including on-line and computer-based learning).

Researchers have also reported their view on RE teaching strategies used in universities. (Nguyen, Armarego, & Swatman, 2002) in their study found that the requirements process, as described in the literature and therefore taught at universities, does not match the real needs of industry. This is because the REE provided to students still includes the traditional teaching methods. However, students do not learn the skills that industry requires through typical lectures. (Callele & Makaroff, 2006) emphasized that the goal of teaching RE is not only to provide students with solid concepts of the subject, but also to expose them to real requirements problems.

(Beatty & Agouridas, 2007) suggested that learning through doing is more effective than learning through being told, and that students should be taught to work on different problematic situations which are similar to those they will encounter in industry.

It can therefore be concluded that educators and students are facing many problems with regard to teaching and studying RE because of the deficiencies in the current approaches to teaching RE. The review and analysis of REE problems conducted in this study also contribute to the list of strategies which have been suggested and compiled by researchers in dealing with RE problems.

2.4 REE problems

Problems in REE are frequently acknowledged within the REE community and reported in several studies, which includes the problems lecturers and students face in teaching and studying RE and the concerns of industry that stem from a lack of RE teaching at universities. In order to extract all the REE problems presented in the literature, a search procedure has been applied to identify studies presenting REE problems, relevant studies have been selected and problems have been extracted from the selected studies. This section presents the search procedure applied, the overview of the selected studies and the problems extracted from selected studies.

2.4.1 Search procedure

In order to extract REE problems from relevant studies, a basic inclusion/exclusion criteria is defined for including studies and then selecting the most related studies for the purpose of data extraction. The basic inclusion criterion is to identify studies related to teaching students RE in universities, REE issues that students and lecturers are dealing with while studying and teaching RE courses, RE training provided to practitioners during their job, and RE issues they are facing due to the lack of REE in universities.

This research will not select studies which are not focused specifically on REE, nor studies which focused on general RE issues that have no impact on REE, nor studies based on SE education issues in general. Searching for the REE problems was accomplished by using several strategies:

- Keywords were formulated and used to search digital libraries. Online search engines such as ACM Digital Library, IEEEXplore Digital Library, Digital Dissertations @ProQuest, ScienceDirect, Springer Link, Google, Google Scholar and Wikipedia were also used, and when relevant results from other digital libraries were found, they were also included;
- In addition to searching digital libraries and using online search engines, the reference lists of relevant articles were also checked in order to extract a complete set of information about the topic.

The articles obtained as a result of the search procedure were read and evaluated according to the aforementioned inclusion and exclusion criteria.

2.4.2 Selected studies

Studies reporting problems in REE were published between 2000 and 2011. Over these 11 years, there was an increase in published papers reporting REE problems, especially from 2005–2009. The increase may be a reflection of the growing awareness of the importance of REE within the SE community. Nevertheless since 2010, the number of studies reporting REE in RE decreased.

From the results of the search, the majority of studies were not validated because they were general. Therefore after data extraction, we did not have a significant number of publications that focused specifically on REE. This led us to the selection of a total of 13 relevant studies that specifically reported on problems in REE. Of the 13 studies selected from the result of the literature search, ten were carried out by researchers from various universities, and three were carried out by industry researchers. The university researchers mostly shared their experiences of teaching RE courses to university students and reported on the REE problems that their students faced. Meanwhile, the industry practitioners proposed approaches to training practitioners, along with presenting RE problems faced by the industry due to a lack of REE in universities. The brief overview of selected studies including introductions of authors and RE course/training they have conducted are presented below.

1) (Zowghi, 2009): The author of this paper is a university researcher who designed and delivered an online postgraduate RE course for the first time to first year Master of Software Engineering students situated in different cities in Iran. The class consisted of 22 students, most of them have had some work experience in software development either working for small companies or contracting independently. They were well aware of the problems and challenges of online education and the reliance on limited information and communication technologies available.

2) (Regev et al., 2009): The authors are university researchers who shared their experience teaching RE as part of an experiential enterprise architecture (EA) course at the Ecole Polytechnique Fe´de´rale de Lausanne (EPFL), Switzerland. The course was the result of a major effort by a teaching team including a professor, four teaching assistants (TAs) and a visiting professor, and uses experiential learning approach for teaching RE. The course uses a low-tech social simulation so that students learn through interaction with real people.

3) (Connor, Buchan, & Petrova, 2009): The authors of this paper are university researchers who have designed and delivered a Masters course to postgraduate students in Software Requirements Engineering (SRE) as part of the Master of Computer & Information Sciences (MCIS) degree at Auckland University of Technology. This course has been designed to overcome some of the issues that have caused the research-practice gap.

4) (Barnes, Gause, & Way, 2008): The authors of this paper are university researchers who have designed and delivered a course to capston senior design project class for an undergraduate degree in Industrial Systems Engineering (ISE) at Binghamton University. This was the second part of a two-semester senior design course. The first semester emphasized requirements elicitation and principles of creative and innovative design. The second semester (and focus of this study) was a capstone project involving analysis of requirements for a given design solution. The application of tools and techniques for the unknown and unknowable of design requirements was tested in a capstone senior design project class.

5) (Hoffmann, 2008): The author is an industry researcher who designed a one hour workshop for anyone having some RE experience who is interested in using theatre techniques for teaching. This workshop was intended to teach RE techniques with communication related issues to help participants to experience the human factor.

6) (Beatty & Agouridas, 2007): The first author is an industry researcher while the second is a university researcher. They have developed and delivered a short eight hours training course to improve upon current industry practices. The course was taught in one day to train the practitioners in the industry.

7) (Smith & Gotel, 2007): The authors are researchers from Pace University New York, who designed and educationally used a board game to introduce small novice organisations with a basic light-weight set of RE good practices.

8) (Huijs, Sikkel, & Wieringa, 2005): The authors are university researchers who designed an approach to integrate RE as an explicit part in different courses in BIT (Business Information Technology) programme at the University of Twente. The programme consists of a 3-year B.Sc. and 2-year M.Sc. programme. It is a joint effort by the Faculties of Computer Science and of Management Science. The BIT curriculum contains regular subjects from both disciplines, as well as courses that aim to integrate both knowledge. Some of these integrative courses are projects, in which student teams address complex case studies or real problems.

9) (Berenbach, 2005): The author is a researcher from industry who is a member of the technical staff at Siemens Corporate Research, and works in the software engineering department. This paper provides an industrial perspective on what college students should be learning about RE at the undergraduate and graduate level, and why they should be learning it.

10) (Jiang et al., 2005): The authors are university researchers who did research on the selection and combination of RE techniques; they also used several case studies which applied the selection process to an industrial software project.

11) (Al-Ani & Yusop, 2004): The authors are university researchers who have designed and delivered a RE course to students at the Faculty of Information Technology in the University of Technology, Sydney. It is a core subject in the Graduate Certificate in Information Technology and Graduate Diploma. The course was offered to both undergraduate and post graduate students. This study reported the experiences of the subject coordinator, lecturer (first author) and the head tutor (second author).

12) (Rosca, 2000): The author is a university researcher who designed and delivered a RE course as a component of the curriculum for the Masters in Software Engineering (MSSE) program, at Monmouth University, New Jersey for a fourteen week semester. In this paper, the author shared his experience related to the active/collaborative approaches in teaching the course.

13) (Gibson, 2000): The author of this paper is a university researcher who designed a course titled Software Engineering (using formal methods) as part of a Master's degree (MSc) in University Henri Poincary (Nancy I), France. The course was taught in 36 hours from which 12 to 16 hours were spent on formal RE (not including their practical coursework).

2.4.3 REE problems extracted from selected studies

In the selected studies, most of the researchers reported a number of REE problems and focused on providing solutions to one or more of these problems, except for three researchers who specifically reported only on their investigated problems; they are (Connor et al., 2009), (Zowghi, 2009) and (Rosca, 2000). The problems which researchers only report but do not address in their studies are referred to as "reported problems". Problems which researchers have addressed and proposed strategies for are referred to as "investigated problems". Table 2.3 presents the reported and investigated problems presented by the researchers. Reported problems are denoted by the letter R, while investigated problems are denoted by the letter I. The last column presents the strategies proposed by the researchers in order to address the investigated problems.

Table 2.3: Reported and investigated problems and proposed strategies from selected studies

No	Researchers	Problems	Proposed strategies
1.	(Regev et al., 2009)	▪ The need to provide practical experience to students in REE (I) ▪ The use of RE in industry is hampered by a poor understanding of its practices and their benefits due to the lack of REE at university level (R) ▪ Students should be provided with experience on the issues found in the workplace, including dealing with ambiguity, uncertainty, confusion, fear, time pressure, collaboration, and corporate politics (R) ▪ Students fail to see the point of spending time on understanding business requirements (R) ▪ Students' lack of awareness on creative techniques and the need to use them to define requirements (R) ▪ Students are taught to define requirements that are complete and rigorous, which is often not the case in organisations (R)	The use of experiential learning approaches using low-tech social simulations

Table 2.3, continued

2.	(Connor et al., 2009)	• The need to overcome the lack of effective communication and light coverage of RE in university programmes that have caused a research-practice gap (I)	Emphasise project-based learning and use lectures, group exercises, online discussions and assignments as pedagogical approaches
3.	(Zowghi, 2009)	• The need to teach RE in order to enable students to overcome the communication barrier between developers and customers, to choose the most effective and suitable analysis and modelling techniques, and to develop necessary skills to produce good-quality RE end products (I)	Use role-play and group work as pedagogical approaches, along with bi-weekly and live lectures
4.	(Barnes et al., 2008)	▪ The need to teach students about knowing the unknown (incomplete requirements) or unknowable (changing requirements) in RE, and how to address this in REE (I) ▪ Newly-graduated engineering students are unable to analyse and structure real-world problems from customers (R) ▪ Most SE projects fail due to requirements because students are not taught to deal with the following RE challenges: o A lack of customer involvement in projects (R) o Incomplete requirements provided by customer (R) o The customer's unrealistic expectations of the project (R) o Changes to requirements and specifications during the project (R) ▪ Identifying and clarifying requirements in a volatile and uncertain environment is one of the challenges involved in teaching students about the realities of RE (R)	Teach RE by offering students a combination of lightweight and easy-to-apply approaches using brainstorming meetings

Table 2.3, continued

5.	(Hoffmann, 2008)	▪ The need to teach RE techniques with soft facts (communication-related issues) (I) ▪ The need to find an unproblematic way to teach students to deal with uncooperative stakeholders while trawling for requirements (R) ▪ Teaching students to understand unspoken issues or subconscious requirements from stakeholders is often easy in theory but difficult in practice (R)	Use improvisation theatre techniques to teach RE
6.	(Beatty & Agouridas, 2007)	• The need to define requirements for the development of RE skills through effective teaching and learning methods (I) ▪ The RE discipline should train students in such a way that they can: o Understand ill-structured problems (R) o Understand environmental behaviours (R) o Learn skills such as conflict resolution, scope defining, facilitating decisions, defining expected system behaviour with a combination of users, system and data states and producing output that is suitable for diverse audience (R)	Design a course covering methods to elicit requirements, with emphasis on facilitated requirements sessions and group dynamics

Table 2.3, continued

7.	(Smith & Gotel, 2007)	▪ The need to teach basic RE good practices to students/novice requirements engineers (I) ▪ Weaknesses in traditional RE approaches that can be addressed through REE include: o Current RE practices failing to keep up with best practices (R) o Requirements problems not being recognised (R) o Incomplete or ambiguous customer requirements (R) o Lack of skills, including insufficient rigor, inadequate development, an overemphasis on functional requirements, perceived impracticability, a lack of awareness, admitting mistakes, selling ideas to management, increased short-term costs and a lack of maturity and guidance (R)	Design an educational board game to teach RE
8.	(Huijs et al., 2005)	▪ The need to provide students with an insight into the importance of RE skills and to improve the quality of the requirements specifications produced (I) ▪ The need to enable students to gain real insight into customer's needs (R)	Integrate RE into several courses and challenge students with authentic cases taken from practice
9.	(Berenbach, 2005)	▪ The need to provide an industrial perspective on what and why students should be learning RE at undergraduate and graduate levels (I) ▪ Lack of interest in RE by students (R) ▪ University faculty not having enough experience or skills to teach human interaction in RE (R)	Provide recommendations to improve the RE curriculum

Table 2.3, continued

10.	(Jiang et al., 2005)	▪ The need to improve poor RE process that results in a lack of REE in most academic programmes by selecting and combining RE techniques based on project characteristics (I) ▪ Lack of REE in most academic programmes, meaning that software developers have to learn RE practices on the job (R)	Use case studies that apply the selection process to an industrial software project
11.	(Al-Ani & Yusop, 2004)	▪ The need to introduce students to the foundations of RE (I) ▪ The importance of the RE phase of systems development must be recognised by academics (R)	Use role-playing and peer assessment as pedagogical approaches
12.	(Gibson, 2000)	▪ The need to examine particular problems in teaching formal RE and teaching students basic RE principles and methods (I) ▪ The step of moving from informal (understanding the problem) to formal (recording this understanding by creating a requirements document) methods is very difficult to learn (and to teach) (R) ▪ Students must be aware of the need for customer involvement during the entire development process (R) ▪ Students must learn techniques to help them cope with changing needs (R)	Use formal methods for teaching RE in several case studies using mathematics and specification notation
13.	(Rosca, 2000)	• The need to introduce students to RE processes, methods and tools (I)	Use role-playing, lectures and laboratory work as pedagogical approaches

Table 2.3 shows that:

1. Many of the extracted problems from different researchers point to similar issues. Therefore in order to avoid redundancy, similar REE problems are grouped together in the next step of the analysis of problems (chapter 3).

2. Several different strategies have been proposed to address the same problem. For example, the problem of introducing students to RE processes, methods and tools was investigated by (Rosca, 2000) and (Gibson, 2000) in 2000 and by (Al-Ani & Yusop, 2004) in 2004. Similarly, the problem of providing industrial experience to students in RE course was investigated by (Berenbach, 2005) in 2005 and by (Regev et al., 2009) in 2009. This shows researchers' continuous efforts in addressing some of the problems.
3. A few problems have been reported by many researchers but have yet to be investigated. For example, the problem of teaching students to structure customer's problems has been reported by (Beatty & Agouridas, 2007), (Barnes et al., 2008), (Gibson, 2000), and (Smith & Gotel, 2007). This shows the need for providing a strategy or solution in future studies.

2.5 Summary

This chapter presented a general introduction to RE, its activities, and its current and recommended offerings in different universities and recommendations by different researchers. The literature review then identified 13 relevant studies presenting REE problems by applying a search procedure. The brief introduction of the authors and purpose of studies have been presented. Finally the problems reported in these studies along with the investigated problems and their proposed strategies have been presented. Further analysis on these REE problems have been performed and presented in the next chapter.

Chapter 3- Formulation of an integrated view of REE problems

3.1 Introduction

This chapter presents detailed analysis and classification of REE problems extracted from the literature and produces an integrated representation of the REE problems along with their relevant information. We call this analysis and presentation of the entire range of REE problems "an integrated view of REE problems." The REE problems identified through literature search have been presented in Chapter 2 and the formulation of an integrated view of these problems is discussed in the following sections.

3.2 Methodology for producing an integrated view of REE problems

This study aims to compile and analyse REE problems, and present them in an integrated representation that allows one to see the whole range of problems together with their related information. In order to serve this purpose, the methodology shown in Figure 3.1 has been developed.

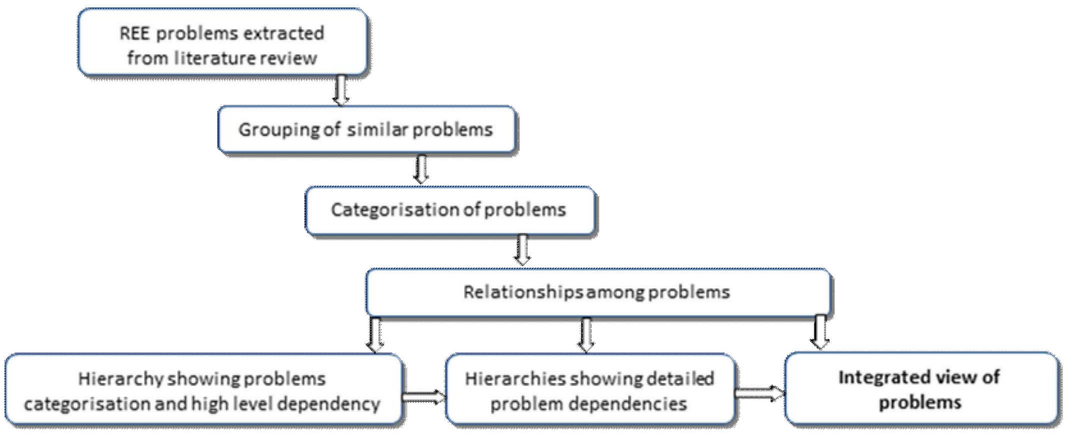


Figure 3.1: Methodology for producing an integrated view of REE problems

This methodology consists of the following steps:

- Extraction of REE problems from literature review: A literature review was carried out in order to identify and extract the main REE problems presented by researchers (the extracted problems have been presented in Chapter 2);

- Grouping of similar problems: Of the identified problems, those related to similar issues were grouped together (this provides the focus for Section 3.3);
- Categorisation of problems: The groups of problems were then divided into categories, depending on the nature of the problem (this provides the focus for Section 3.4);
- Relationships between problems: The relationships between groups of problems were established and presented as:
    - A hierarchy showing categorisation and high-level dependencies (this provides the focus for Section 3.5);
    - Hierarchies showing detailed problem dependencies (this provides the focus for Section 3.6);
    - Representation of an integrated view of problems (this provides the focus for Section 3.7).

The above methodology performs a detailed analysis and classification of REE problems extracted from the literature, and presents the groupings and classification of the problems and their relationships in an integrated view.

3.3 Grouping of similar problems

In this section, similar REE problems are grouped together and a frequency is assigned to each group. For example, the group entitled "Lack of understanding of RE techniques" is assigned a frequency of six, which means that the problem has been identified by six different researchers. This gives an indication that the problem is commonly researched and reflects the degree of its importance. Table 3.1 summarises the groups of problems as reported by the researchers (refer to Table 2.3), combined under one general heading and organised in ascending order of their frequency. The reported and investigated problems are denoted by R and I respectively.

Table 3.1: Groups of problems

No	Problem	Frequency
1)	**A lack of understanding of RE techniques**	
	▪ Use of RE in the industry is hampered by poor understanding of its practices and their benefits due to a lack of REE at university level (R)	6
	▪ Selection of techniques that are ill-suited to a particular project (I)	
	▪ Current RE practices fail to keep up with best practices (R)	
	▪ The ability to choose the most effective analysis and modelling techniques with which to solve the problem at hand (I)	
	▪ Teaching basic RE good practices to students/novice requirements engineers (I)	
	▪ Students' lack of awareness of creative techniques and the need to use them to define requirement (R)	
2)	**Teaching communication skills**	
	▪ Teaching RE techniques with communication-related issues (I)	**5**
	▪ Providing students with an insight into the importance of requirements analysis and communication skills (I)	
	▪ Communication barriers between developers and customers (I)	
	▪ Lack of effective communication (I)	
	▪ The university faculty not having enough experience or skills to teach human interaction in RE (R)	
3)	**Teaching students to analyse and structure real-world problems from customers**	
	▪ Newly-graduated engineering school students are unable to analyse and structure real-world problems from customers (R)	4
	▪ Understanding ill-structured problems (R)	
	▪ Requirements problems going unrecognised (R)	
	▪ The step of moving from informal (understanding the problem) to formal (recording this understanding by creating a requirements document) methods is very difficult to learn (and to teach) (R)	

Table 3.1, continued

4)	**Dealing with incomplete requirements**	
	▪ Incomplete requirements provided by customer (I)	3
	▪ Students are taught to define requirements that are complete and rigorous, which is often not the case in organisations (R)	
	▪ Incomplete or ambiguous customer requirements (R)	
5)	**A lack of customer involvement**	
	▪ Students must be aware of the need for customer involvement during the entire development process (R)	3
	▪ Lack of customer involvement in projects (R)	
	▪ The need to find an unproblematic way to teach students to deal with uncooperative stakeholders while trawling for requirements (R)	
6)	**Dealing with unrealistic customer expectations**	
	▪ Unrealistic customer expectations of the project (R)	3
	▪ Understanding unspoken issues or subconscious requirements from stakeholders (R)	
	▪ Enabling students to obtain real insights into customer's needs (R)	
7)	**Dealing with changing requirements**	
	▪ Students must learn techniques to help them cope with changing needs (R)	3
	▪ Changes to requirements and specifications during the project (R)	
	▪ Managing changing requirements and the uncertainty of all associated unknowns (incomplete requirements) or unknowables (changing requirements) (I)	
8)	**Students need to understand the importance of RE**	
	▪ Students fail to see the point in spending time on understanding business requirements (R)	3
	▪ The importance of RE phase of systems development must be recognised by academics (R)	
	▪ Lack of interest in RE by students (R)	

Table 3.1, continued

9)	**Dealing with RE challenges**	
	▪ Identifying and clarifying requirements in a volatile and uncertain environment is one of the challenges of teaching students about the realities of RE (R)	3
	▪ Understanding environmental behaviours (R)	
	▪ Students should be provided with experience on issues found in the workplace, including dealing with ambiguity, uncertainty, confusion, fear, time pressure, collaboration and corporate politics (R)	
10)	**Teaching basic RE concepts and tools**	
	▪ Teaching basic RE activities, methods and tools which are available for eliciting, analysing, specifying, validating and managing requirements (I)	3
	▪ Introducing students to the foundations of RE (i.e. front-end activities, methods, techniques and tools) (I)	
	▪ Teaching students basic RE principles and methods (I)	
11)	**Lack of RE skills**	
	▪ Lack of skills such as conflict resolution, scope definition, facilitating decisions, defining expected system behaviour with a combination of users, system and data states and producing output which is suitable for a diverse audience (R)	3
	▪ Lack of skills, including insufficient rigor, inadequate development, an overemphasis on functional requirements, perceived impracticability, a lack of awareness, admitting mistakes, selling ideas to management, increased short-term costs and lack of maturity and guidance (R)	
	▪ Defining requirements for the development of RE skills through effective teaching and learning methods (I)	

Table 3.1, continued

12)	**Providing industrial experience in REE**	
	▪ Teaching RE in order to improve upon current industry practices (I)	2
	▪ Providing an industrial perspective on what and why students should be learning about RE at undergraduate and graduate levels (I)	
13)	**Light coverage of RE material in university programmes**	
	▪ Relatively light coverage of RE in universities (I)	2
	▪ Lack of REE in most academic programmes, meaning that software developers have to learn RE practices on the job (R)	
14)	**Teaching skills in order to produce good-quality requirements specifications**	
	▪ Poor quality of RE end products (I)	2
	▪ Teaching students to improve the quality of the requirements specifications they produce (I)	

Each group of problems refers to some specific issue that causes problems for students and educators in universities, and practitioners in industries. These issues are discussed below.

- **A lack of understanding of RE techniques**

There are a number of RE techniques available and practitioners have to choose one or a combination of these techniques based on the characteristics of a project. However, practitioners are usually unable to understand these techniques and their particular use, which results in the selection of a technique ill-suited for a particular project. This is an important problem and is reported by many researchers because the successful completion of the RE process depends on the right selection of techniques used.

- **Teaching communication skills**

Requirements elicitation is the first and important step of the RE process. To be able to perform this process efficiently, a requirements engineer must be equipped with the necessary communication skills. He must be able to communicate with a wide range of people from different backgrounds and goals, and should be able to ask them the right questions at the right time in order to capture their real needs. Unfortunately as reported by the researchers, there is a lack of communication skills that is required in order to perform well during the elicitation process which has led to a communication barrier between the developers and customers. Teaching these skills is very important but difficult to do; students need to understand the importance of these skills and should be aware of the need for collaboration in RE.

- **Teaching students to analyse and structure real-world problems from customers**

The problems stated by customers are usually incomplete and ill-structured. Analysing and structuring these problems before solving them is an important RE issue, but has been given less attention when teaching RE. As a result, newly graduated engineers are unable to understand the informal representation of problems and to structure these problems from the real-world mess. Therefore there is a need to explicitly teach this issue in REE.

- **Dealing with incomplete requirements**

The requirements elicited from stakeholders are usually incomplete. Therefore, it is challenging for practitioners to understand and work on these incomplete requirements. This problem is very important and is reported by many researchers in RE literature. Thus, there is a need for practitioners to be able to deal with incomplete requirements.

- **A lack of customer involvement**

RE is a phase that requires customers to be involved during the whole RE process but this is not usually the case. A customer is either not always there to answer questions or he is not always cooperative. Therefore, practitioners should have the skills to get the customers involved in the project and give it the appropriate attention needed.

- **Dealing with unrealistic customer expectations**

During requirements elicitation, practitioners always try to get as many requirements as possible as stakeholder on the other hand, might omit those requirements that are too obvious in his opinion. Also, often the requirements provided by stakeholders are unrealistic and difficult to implement within the provided time and resources. Understanding these unspoken issues and unrealistic customer expectations proves very difficult for practitioners, thus they should have the necessary skills to deal with them.

- **Dealing with changing requirements**

During the software development process, the customers' requirements keep on changing and it becomes difficult for practitioners to deal with these changing requirements as the project moves from the initial to the later stages of development. Therefore, the practitioners must be equipped with the necessary skills to deal with the changing requirements.

- **Students need to understand the importance of RE**

Students see RE as a boring subject. They prefer to work as a software developer rather than an analyst or a requirements engineer. However, students should recognize the importance of the RE phase of software development, while educators should find the means to engage them in the learning process. Educators should emphasize on practice to achieve the desired practical work in RE because students do not feel that RE is an important phase and they fail to see the point in spending much time in order to understand the business requirements. Also, the course should be designed in a way that develops students' interest in RE.

- **Dealing with RE challenges**

During product development, practitioners have to deal with a number of RE challenges such as ambiguity, uncertainty, confusion, fear, time pressure, collaboration, and corporate politics, along with the issues discussed above. Therefore the practitioners must be able to deal with these challenges.

- **Teaching basic RE concepts and tools**

The primary purpose of teaching RE is to introduce students to the process of RE, and to the methods and tools available for eliciting, analysing, specifying, validating and managing requirements. In the literature, the researchers have presented the problems that students and lecturers faced in studying and teaching basic RE skills. These problems need to be addressed in order to provide students with the foundations of RE before they are taught other RE skills.

- **Lack of RE skills**

Due to the lack of REE in universities, students are left with a lack of RE skills. The literature presented many of the skills that are lacking such as the practitioners ability to resolve conflicts, define scope, facilitate decisions, define expected system behaviour in a combination of user, systems and data states, and produce outputs suitable for a diverse audience, insufficient rigor, perceived impracticability, lack of awareness, increased short-term cost, current practices lag best practices, and lack of maturity and guidance These skills needs to be provided in REE.

- **Providing industrial experience in REE**

RE is the most important and difficult phase of software development life cycle, but students cannot understand its importance. They may not be able to cope with large scale software development until and unless they are provided with the organizational experience in REE. This aspect of REE is very important and researchers emphasized that REE should be relevant to industrial practices so that the students may be able to cope with the challenges related to software development.

- **Light coverage of RE material in university programs**

In software engineering programs in universities, RE is taught as a complete and independent course. Unfortunately, due to a lack of proper course outlines and practical experiences, RE is not taught in depth. Hence, students have only some vague knowledge through lectures. Due to this lack in REE, software developers have to learn RE practices on the job which results in many problems. Therefore, proper course outlines, skills and resources are needed to teach students the required RE concepts, techniques and skills to enable them to become good requirements engineers in the near future.

- **Teaching the skills to produce good quality requirements specifications:**

RE is the first phase of software development cycle and the requirements specifications is the final output of the RE process. It is the first point of reference for the following activities in the development cycle. The problem facing practitioners is the poor quality of the end product such as missing or ambiguously presented or misinterpreted information, poor representation, untestable statements of requirements, and redundant information. In REE, students should be provided the skills and practice of writing good quality requirements specifications. The students must learn not only to write good quality specifications but they should also be able to use and update it at later development stages.

3.4 Categorisation of problems

From this point onwards, each group will be dealt as a single problem. It has been observed that these groups of problems relate to two different factors; some of these problems relate to the RE curriculum (namely issues that cause problems for students and educators in learning and teaching RE), while others are related to RE practices (namely issues that cause problems for practitioners). Therefore, these problems have been further categorised into two categories that are discussed below.

3.4.1 Problems relating to the RE curriculum (REc)

This category includes problems that are specifically related to universities, such as the topics that university programmes lack, the issues that university programmes should pay attention to, the problems faced by students and educators, what should be included in RE courses. The problems in this category are denoted by "REc". It was observed that the problems in this category are either related to the deficiencies in RE courses such as the light coverage of RE material in university programmes, or related to the problems experienced by educators and students regarding teaching and studying such as teaching students to structure problems from realistic dilemmas. Once again, this represents two different perspectives, therefore further categorisation was needed. Thus, these problems were further subdivided into two categories which are discussed below.

1) Deficiencies in RE courses (REc-1)

Educators at universities design RE programmes to be taught to students, but deficiencies in the course design (e.g. the course may not cover the necessary details, or may fail to develop students' interest in the subject) may lead to many problems for those students who later work as practitioners in the RE industry. Issues that university faculties and educators should consider while designing and delivering RE courses are placed in this category and denoted by 'REc-1'.

2) Difficulties experienced in teaching RE issues (REc-2)

RE courses include the concepts, tools, techniques and skills that should be taught by educators in universities, but because of the theoretical nature of RE, it is very difficult to effectively teach these issues to students. This requires teaching methods that can help teach problematic RE issues. This category therefore includes the RE issues that are currently being taught in RE courses, but which educators still find difficult to teach, while students face difficulties in understanding them. These issues are denoted by 'REc-2'.

3.4.2 Problems relating to RE practice (REp)

This category includes problems that practitioners often face while performing RE in the industry, such as dealing with unrealistic customer expectations, incomplete requirements and the inability to select techniques which are best suited to a particular project. The problems in this category are denoted by 'REp'.

It has been noted that these problems can also be presented by two different perspectives, as some practitioners face challenges due to the lack of REE at universities such as a lack of requirements elicitation skills, while others face problems due to the problematic nature of RE such as dealing with changing requirements. Therefore, these problems have been further subdivided into two categories as discussed below.

1) Lack of RE skills (REp-1)

When RE courses in universities fail to cover topics in sufficient detail, then students are left lacking the skills required to perform the RE process in practice. This category includes such problems, which are denoted by 'REp-1'.

2) RE challenges (REp-2)

While working on real projects, practitioners, in particular junior practitioners, may face many RE challenges and may be unable to deal with some of them. The inability to deal with these challenges can be due to a lack of REE in universities. This category includes such challenges, which are denoted by 'REp-2'.

3.5 Presenting problem categorisation in a hierarchy

The categorisation of the problems discussed above has led to the Problem Hierarchy illustrated in Fig. 3.2, which provides a visual representation of the categories (REc, REp), sub-categories (REc-1, REc-2, REp-1, REp-2) and the RE problems in each sub-category, as well as the high-level problem dependency.

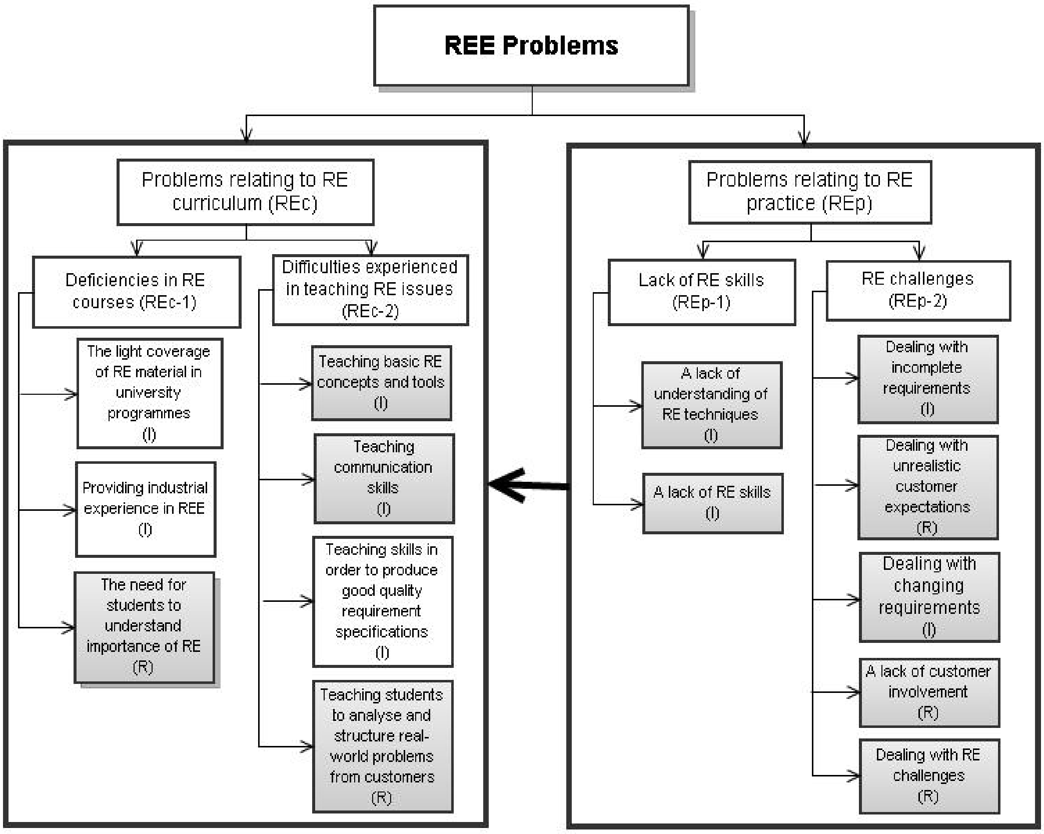


Figure 3.2: The hierarchy showing problem categorisation and high-level dependency

The problem categorisation is linked by normal (inner) lines, whereas the two bold lines separating the problem categories, with an arrow pointing from the REp problem box to the REc problem box, show the dependency of REp problems on REc problems. This dependency has been drawn based on the fact that problems in RE practice occur due to the lack of REE at university. The reported and investigated problems tabulated in Table 3.1 (denoted R and I) as well as the frequencies of the problems, are also integrated into this hierarchical diagram. The shaded boxes represent problems with frequencies of three, four or five, whilst non-shaded boxes represent problems with a frequency of two.

3.6 Presenting detailed dependencies between problems

From a thorough, detailed analysis of the problems, it has been observed that the dependency between the two problem categories is related to the fact that each REp problem is linked to one or more REc problems. Some problems are dependent on others, while some are as a consequence of others. For example, the problem of the lack of requirements elicitation skills and the lack of understanding of RE techniques can occur due to the light coverage of RE material in university programmes. If the problem of providing industrial experience in REE is addressed, then students can learn to deal with incomplete and changing requirements. Therefore, both these REp problems are dependent on the REc problem of providing industrial experience in REE. A significant number of these types of dependencies exist between both categories. The dependencies of REp problems on each REc problem are discussed below.

1) The light coverage of RE material in university programmes: Three REp problems are dependent upon this problem, which are: the lack of understanding of RE techniques; the lack of RE skills; and dealing with RE challenges. The first two problems may be the result of the problem of light coverage of RE material, as practitioners can suffer from a lack of skills if they have not studied RE in depth at university.

The third dependent problem, on the other hand, deals with RE challenges, as practitioners may be able to deal with RE challenges if they have been taught the RE material in sufficient detail at university. Therefore, these problems can be addressed by solving the problem of light coverage or RE material in university programs.

2) Providing industrial experience in REE: Five REp problems are dependent upon this problem, which are: dealing with changing requirements; dealing with unrealistic customer expectations; a lack of customer involvement; dealing with incomplete requirements; and dealing with RE challenges. Practitioners may be able to deal with these RE challenges if they have been provided with industrial experience during their REE. Therefore, these five problems can be reduced if the problem of providing industrial experience in REE is addressed.

3) The need for students to understand the importance of RE: Two REp problems are dependent upon this problem, namely the lack of understanding of RE techniques and the lack of RE skills. One reason for this lack of skills can be that students do not see RE as an important process or they perceive RE as a boring subject and do not give due attention when learning these skills while studying RE. Students can be motivated to learn these skills by making them understand the importance of RE. Therefore, these problems can be addressed through finding a solution to the students' problem.

4) Teaching basic RE concepts and tools: Two REp problems are dependent upon this problem, namely the lack of understanding of RE techniques and the lack of RE skills. Students may lack these skills if they are not taught basic RE concepts and tools to a sufficient extent. Therefore, these problems can be addressed by tackling the problem of teaching basic RE concepts.

5) Teaching communication skills: One REp problem is dependent upon this problem, which is the lack of customer involvement. Students can be taught to deal with customers by being taught communication skills because it involves collaborating with the customer and other requirements elicitation skills. Therefore, this problem can be addressed by confronting the issue of teaching communication skills.

6) Teaching the skills in order to produce good-quality requirement specifications: Two REp problems are dependent upon this problem, namely the lack of understanding of RE techniques and the lack of other RE skills. Students can learn these skills while learning to produce good-quality requirements specifications because producing good-quality requirements specifications involves the selection of appropriate RE techniques and the necessary RE skills. Therefore, these problems can be addressed by managing the problem of teaching requirements specifications.

7) Teaching students to analyse and structure real-world problems from customers: Two REp problems are dependent upon this problem, which are: the lack of customer involvement; and dealing with unrealistic customer expectations. If students are taught to analyse and structure real-world problems from customers, they will also learn to deal with these two challenges as they must involve the customers in the project in order to ascertain their requirements, and they have to understand the customer's expectations of the project in order to analyse and structure those requirements. Therefore these problems can be addressed by considering the problem of teaching students to analyse and structure real-world problems from the customers.

Table 3.2 summarises the number of specific REp problems that are dependent on REc problems.

Table 3.2: REp problem dependency

No.	REc problems	REp-dependent problems
1.	The light coverage of RE material in university programmes	▪ A lack of understanding of RE techniques ▪ A lack of RE skills ▪ Dealing with RE challenges
2.	Providing industrial experience in REE	▪ Dealing with changing requirements ▪ Dealing with unrealistic customer expectations ▪ A lack of customer involvement ▪ Dealing with incomplete requirements ▪ Dealing with RE challenges
3.	The need for students to understand the importance of RE	▪ A lack of understanding of RE techniques ▪ A lack of other RE skills
4.	Teaching basic RE concepts and tools	▪ A lack of understanding of RE techniques ▪ A lack of other RE skills
5.	Teaching communication skills	▪ A lack of customer involvement
6.	Teaching the skills in order to produce good-quality requirement specifications	▪ A lack of understanding of RE techniques ▪ A lack of other RE skills
7.	Teaching students to analyse and structure real-world problems from customers	▪ A lack of customer involvement ▪ Dealing with unrealistic customer expectations

In order to visualise the details of these dependencies alongside the problem categorisations, the hierarchy of problems (Fig. 3.2) has been extended to incorporate these detailed dependencies. Fig. 3.3a shows the dependency of REp-1 problems on REc problems, while Fig. 3.3b shows the dependency of REp-2 problems on REc problems.

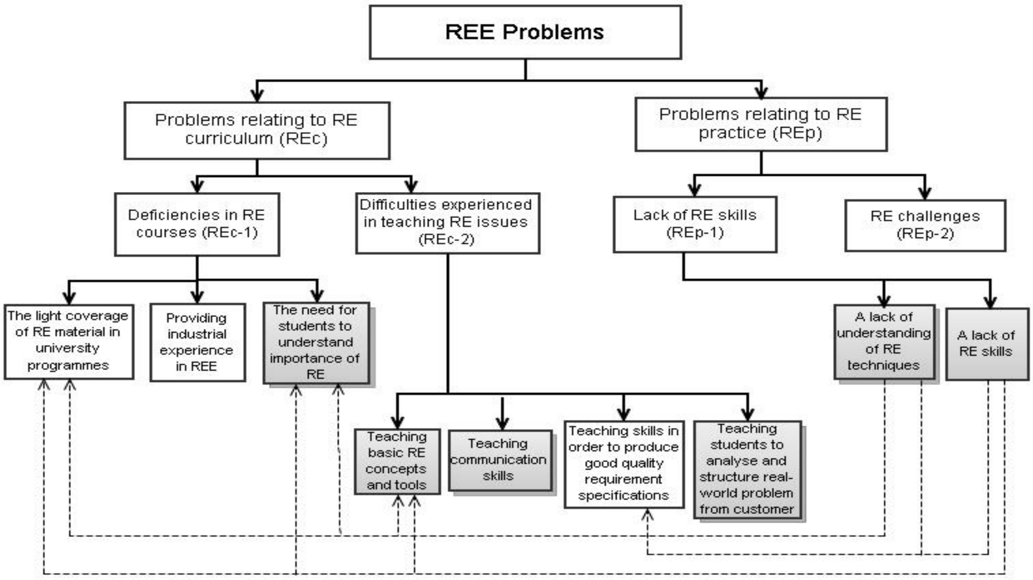


Figure 3.3a: The dependency of REp-1 problems on REc problems

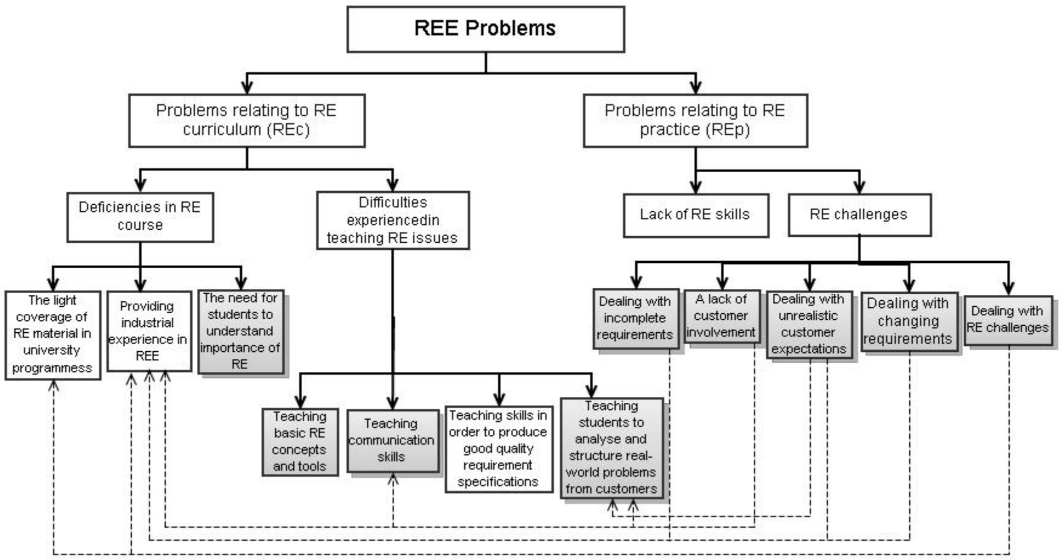


Figure 3.3b: The dependency of REp-2 problems on REc problems

In these figures, the dependencies between the individual problems are shown by dotted lines with arrow heads pointing from the REp problems to the REc problems, showing that REp problems are dependent upon REc problems. The shaded boxes represent the problems with frequencies of three, four, five or six, while the non-shaded boxes represent problems with a frequency of two.

3.7 Integrated view of problems

The results from Sections 3.3 to 3.6 are collected and portrayed in a view called the "integrated view of REE problems", shown in Fig. 3.4. The information presented in the integrated view is summarised below:

- **Problem types:** The problem types that are reported and investigated (as presented in the previous chapter) are presented by linking investigated problems to a box labelled "investigated", and reported problems to a box labelled "reported."
- **Frequencies:** The information of frequencies of groups (as presented in Section 3.3) is presented using shaded and non-shaded ovals. The problems with frequencies of three, four, five and six are shown in shaded ovals, while the non-shaded ovals represent problems with a frequency of two.
- **Problem categorisation:** The two problem categories (as presented in Section 3.4 and Section 3.5) are presented using two large ovals, and the small ovals identify the problems within these categories.
- **High-level dependency:** The high-level dependency of REp problems upon REc problems (as presented in Section 3.5) is presented by an arrow pointing from the REp problems oval to the REc problems oval.
- **Detailed dependencies and their types:** The detailed dependencies of each REp problem on one or more REc problems, and the types of these dependencies (as presented in Section 3.6), are presented using diamond-shaped boxes pointing towards the REc problems, identifying which REp problems are dependent on each of the REc problem and in what way they are dependent.

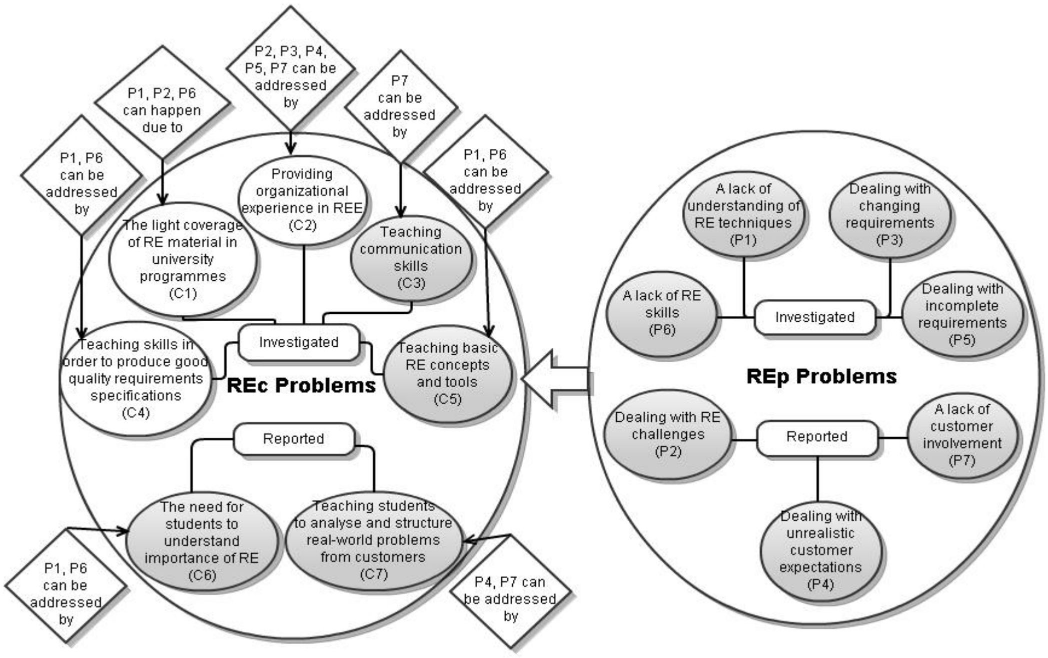


Figure 3.4: An integrated view of REE problems

3.8 Summary

This chapter presented the analysis of REE problems in order to produce an integrated representation. Through literature search, the main problems associated with REE were identified and extracted from studies previously performed by researchers. A detailed analysis of the problems was performed in which they were arranged into groups with similar issues, classified into different categories and the relationships between them identified. Finally, an integrated representation was produced that provides an overview of relevant information on the REE problems. The next chapter will present the investigations performed among students and lecturers in order to verify the problems presented in the integrated view and formulate the research focus.

Chapter 4- Formulating the research focus through investigations and analysis

4.1 Introduction

This chapter verifies the problems presented in the integrated view and formulates the focus for this research by performing investigations on lecturers and students about the problems presented in the integrated view and analysing the results.

4.2 Methodology

The study aims at verifying the problems presented in the integrated view and formulating the research focus that contributes in addressing one of the major problems identified as the research gap. In order to achieve this aim, the methodology shown in Figure 4.1 is used.

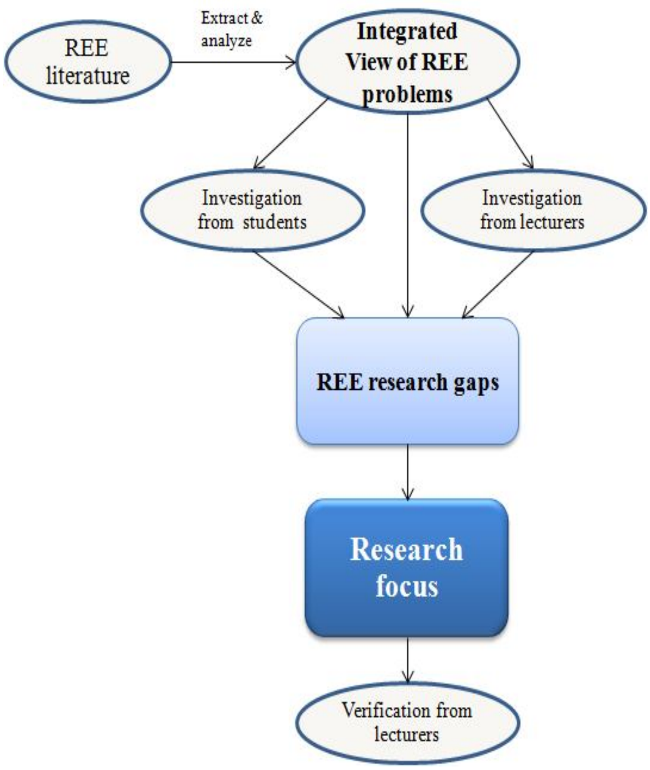


Figure 4.1: Methodology for formulating the research focus

A combination of data gathering and analysis techniques were applied to verify the REE problems presented in the integrated view and to identify the research focus. The two principle sources of information were the literature and the two investigations performed on students and lecturers.

The integrated view was produced by extracting and analysing the REE problems presented in the literature. The problems presented in the integrated view were then verified by the investigations performed on software engineering undergraduates who have studied RE and lecturers with the experience of teaching RE for several years. The investigation results were analysed with reference to the integrated view of problems and the REE research gaps were identified, followed by a selection of one problem as the research focus to be addressed further. The selected focus was then supplemented by another investigation performed on the lecturers to gather suggestions and recommendations on the selected problem.

4.3 Investigations

In order to verify the REE problems presented in the integrated view, investigations on students and lecturers were performed through questionnaire surveys. The investigations were aimed at undergraduate software engineering students who have taken RE as a course, and lecturers and RE researchers who have experience in teaching RE course or have research experience in RE along with teaching RE. To define the goal of the study, Goal Question Metric (GQM) approach was used. GQM is an approach developed in response to the need for a goal-oriented approach that would support the measurement of processes and products in software engineering domain. If viewed narrowly, GQM approach may be seen as purely an approach for choosing metrics (Differding, Hoisl, & Lott, 1996). GQM approach is based upon the assumption that for an organization to measure in a purposeful way it must first specify the goals for itself and its projects, then it must trace those goals to the data that are intended to define those goals operationally, and finally provide a framework for interpreting the data with respect to the stated goals (Caldiera & Rombach, 1994). The goal of this study, defined using GQM template is presented in table 4.1.

Table 4.1: GQM defining goal of the investigation study

Analyse	REE problems
For the purpose of	Verification
With respect to	Integrated view
From the point of view of	Researchers
In the context of	Software engineering undergraduates who have studied RE and lecturers who have experience in teaching RE.

This section describes the questionnaires used for performing investigations, the procedure applied and the data analysis methods.

4.3.1 Questionnaires

Based on the experience of students as well as lecturers, it was felt necessary to produce two separate questionnaires; students' questionnaire and lecturer's questionnaire. The questionnaires include quantitative as well as qualitative questions. The complete students' questionnaire is included in Appendix A while the lecturers' questionnaire is included in Appendix B.

The questions in the questionnaires are mostly based on the problems presented in the integrated view. The problems in the REc category are referred to as RE elements and those in the REp category are referred to as RE challenges in the questionnaires. They are investigated separately due to the difference in the nature of these problems. However, collectively, all problems are called RE issues.

It was also felt necessary to understand the difficulties that students and lecturers are facing whilst studying RE in order to compare their opinions with those presented in the integrated view and also to gather their suggestions for improving the course, which will help identify the problems that need to be investigated further. Therefore, the students' and lecturers problems and suggestions were asked separately using two open questions. The questionnaires were divided into the following three parts.

1. RE elements and challenges were listed and
    - Students were asked whether these were taught to them in class. If they were, could the students sufficiently perform them in real projects with a rating level of 1–5 (1=Yes, very sufficient; 5=No).
    - Lecturers were asked whether these have been emphasized in the RE syllabus they have taught with a rating level of 1-4 (1=highly emphasized, 4= not emphasized). They were asked to give reasons if any of the RE elements or challenges was given less emphasis or was not emphasized in class.
2. RE issues were listed and
    - Students were asked to highlight those they found difficult to understand.
    - Lecturers were asked which of the RE issues they found most difficult to teach and why.
3. REE problems and suggestions
    - Students were asked to state the problems they faced during the RE course and their suggestions for improving the course
    - Lecturers were asked about the problems they faced while teaching RE and their suggestions for improving the RE course

4.3.2 Participants

The students as well as lecturers participated in the investigations. The participant's detail is presented below.

4.3.2.1 Students

For the first investigation, software engineering undergraduates who have studied and passed RE course were selected as subjects for this study. These students were selected based on the fact that in most Bachelor of Science in Software Engineering (BSc SE) programmes, RE is offered as a core module. As a core module, the course normally covers major aspects of RE, therefore, it is assumed that these students can better respond to the questions related to REE problems.

A total of eighty nine undergraduate students participated in the study which consisted of two groups. One group consisted of 45 students from the University of Malaya (UM), Malaysia. We selected all the students in two available classes who have studied RE course. Another group consisted of 44 students from Mehran University of Engineering & Technology (MUET) and University of Sindh (US), Pakistan. These students volunteered themselves by responding to email invitations and class announcements.

4.3.2.2 Lecturers

The second investigation is aimed at lecturers who have taught RE course. 18 lecturers participated in the study of which 7 are from Malaysia, 5 are from Pakistan and 6 are RE researchers who are also teaching RE in institutions from different countries. The respondents have teaching experience of 5-20 years.

4.3.3 Procedure

4.3.3.1 Student's investigation

The questionnaire was distributed amongst the students from UM, and it took around two weeks to complete the study. While an online questionnaire was made available to the students in Pakistan and it took around 3–4 weeks to receive their feedback. The average time to complete the questionnaire was 20 min.

4.3.3.2 Lecturer's investigation

An online questionnaire was made available and lecturers were sent request through emails to fill in the questionnaire.

4.3.4 Data analysis methods

The investigations were performed using questionnaires which include qualitative as well as quantitative questions. Descriptive analysis was used to analyse the quantitative data. For the analysis of the qualitative data, an approach based on grounded theory (Biasutti, 2011) was adopted as the theoretical framework. An inductive method based on "constant comparative method" (Strauss & Corbin, 1998) was employed to analyse and to categorise the two open questions. One of the main characteristics of this inductive approach is that the categories emerge from the data by the use of an inductive analysis rather than coding of data according to prearranged categories (Charmaz & Henwood, 2008). The constant comparative method consists of the following five phases.

1) Immersion: All detectably different answers are recognised.
2) Categorisation: Detectably different answers are divided into categories.
3) Phenomenological reduction: Themes emerge from the categories.
4) Triangulation: The quotes are used to support the researcher's interpretation.

5) Interpretation: Researcher’s interpretation based on supporting quotes is presented, and a complete explanation of outcomes carried out in connection to previous research and/or models. (Biasutti, 2011)

This method of analysis has been fruitfully adopted in earlier research examining online music learning (Seddon & Biasutti, 2009) and an e-learning as a university module (Biasutti, 2011).

4.4 Investigations Results

The following sections present the results of both investigations and analysis.

4.4.1 Students’ investigation results and analysis

In the students’ questionnaire, the first two parts are quantitative (consists of closed questions) and the third part is qualitative (consists of open questions). The data analysis and results of the three sections are described below.

4.4.1.1 RE elements and challenges

The students were taught RE course by means of lectures (selected by all the students), labs (63%), presentations (58%) and group discussion (41%). A few students mentioned other approaches, namely class assignments, tutorials and quizzes. The students’ response to the first part, that is about RE elements and challenges is shown in Figure 4.2.

The results showed that RE elements scored a mean of 2.52, while RE challenges scored a mean of 2.83 for all recipients out of 89. This shows that on average, for the RE elements, students’ responses are more towards the positive side (“yes, very sufficient” and “yes, sufficient”) than the negative side (“yes, not very sufficient”, “yes, insufficient” and “No”), whereas for the RE challenges, responses veered more towards the negative side rather than the positive side.

It can be observed from the results that almost all the students reported being taught RE elements in the class, and on average, more than half felt they had sufficient skills to perform in real projects whilst the rest did not believe they were sufficiently equipped with these skills. On the other hand, around a quarter of students reported that they had not been taught to deal with RE challenges. On average half felt they had sufficient skills, whilst the rest felt they did not have sufficient skills to face the RE challenges of real projects.

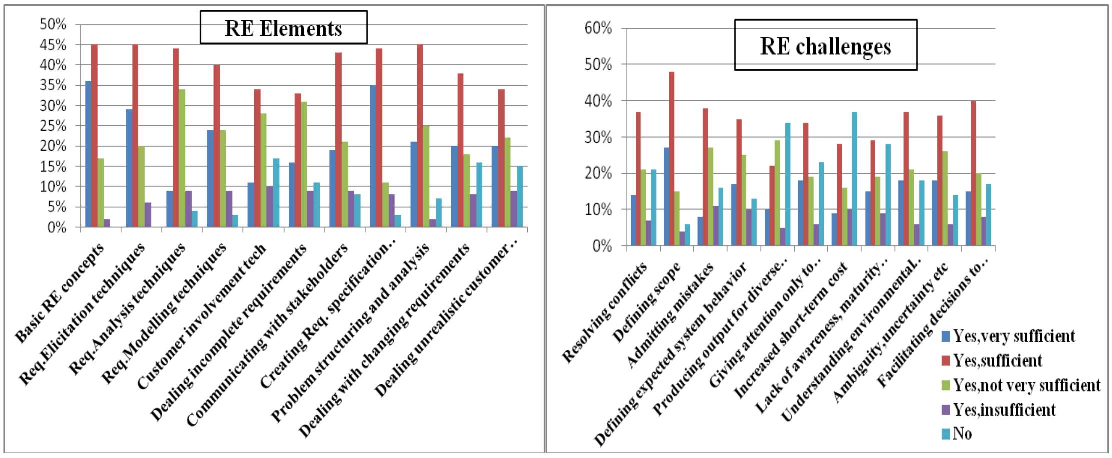


Figure 4.2: Students’ responses to RE elements and challenges

4.4.1.2 RE issues

The students’ response to the second part that is about RE issues is shown in Figure 4.3. From the results, it can be observed that students had difficulties in understanding many issues. The issues that they selected were working on RE tools, problem structuring and analysis, and dealing with changing requirements, incomplete requirements and customers’ unrealistic expectations. However, many students also found other issues difficult to understand.

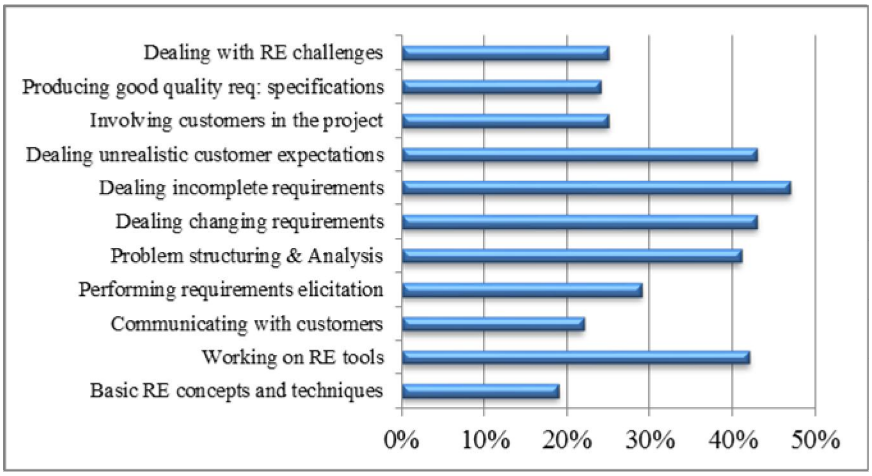


Figure 4.3: Students’ responses to issues they found difficult to understand

4.4.1.3 REE problems and suggestions

The third part of the questionnaire consists of two open questions. A constant comparison method was applied to the results of these questions.

The results of the first open question (“Which problems do you faced during the RE course”) are presented below. Figure 4.4 reports the first three steps of the quantitative data while Table 4.2 reports the results of the next two steps. In the immersion phase, the answers to the question were read and 64 different answers were identified. Then, similar answers were grouped together into 20 categories in the categorisation phase. In the phenomenological reduction phase, five themes emerged, which were understanding RE concepts, working on RE activities, lack of practical work, working on RE tools and facing RE challenges. In the triangulation phase, quotes from the answers to the questions were used to support the interpretation of themes.

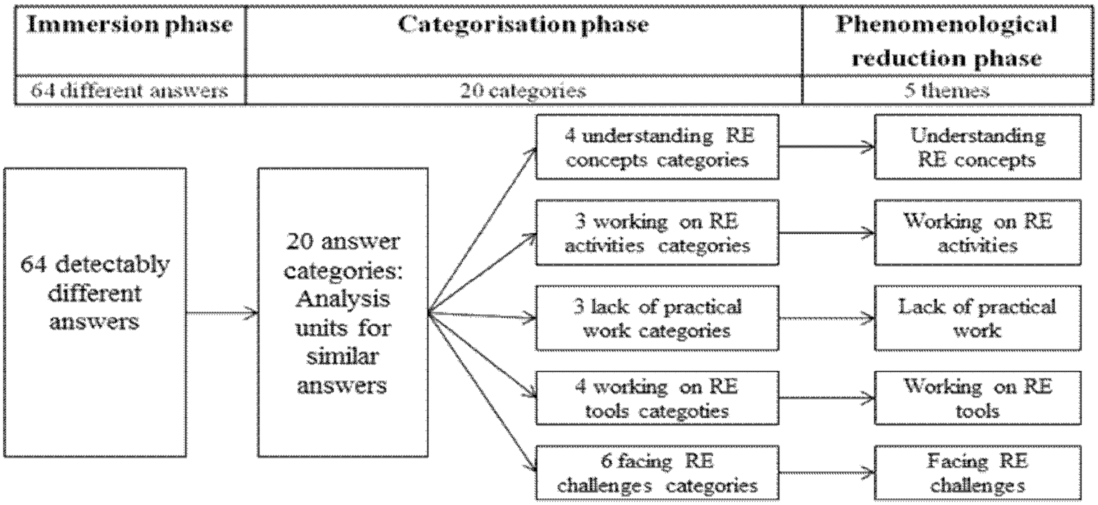


Figure 4.4: A diagram of first three steps of the inductive analysis for the qualitative part of the questionnaire (problems in RE course taught in universities)

Table 4.2: Triangulation phase: Supporting quotes for the five themes of the problems in RE course taught in universities extracted from the answers given by participants

Themes	Supporting quotes	Interpretation
Understanding RE concepts	"There is insufficient information provided during the course". "Difficult to understand information on RE, not fully understand what lecturer teaches". "Hard to understand the concept". "The theories are quite boring and difficult to remember".	These quotes support the reported problems faced by the participants in understanding RE concepts due to the way the RE course had been taught to them.
Working on RE activities	"Difficulty in eliciting requirements". "Not clear about the procedure to elicit or analyse requirements". "Requirements documentation is difficult and troublesome". "Understanding, analysing and structuring initially-presented customer requirements".	These quotes support the reported problems faced by participants whilst studying requirements elicitation, analysis and documentation.

Table 4.2, continued

Lack of practical work	"All theory, not much practice, and no involvement/experience of real world projects". "All we have studied is theory; there is a lack of practical experience". "Lack of practical training, all is based on theory". "It would have been better if taught in a practical way, but it was completely theoretical. So it was very difficult to understand the customer, user and requirements".	These quotes support the reported problems faced by participants due to a lack of practical work and implementation of the RE concepts that they had been taught.
Working on RE tools	"Superficial exposure to RE tools". "The creation of package, documents is not very clear as we had to learn it by ourselves. We need guidance in order to learn the tool, which is difficult to explore". "Difficult to use RE tools – tools are too complex to learn". "The basic problem I faced is the right implementation of the tools used for requirement engineering".	These quotes support the reported problems faced by participants whilst learning and using RE tools.

Table 4.2, continued

Facing RE challenges	"I have difficulties in understanding certain concepts in requirements elicitation, managing the changing requirements of stakeholders, and applying techniques to trace the problems". "Uncertain about whether the requirements that I have elicited are correct or complete or not". **"**Problem structuring and dealing with customers' changing and incomplete requirements are very difficult". "Facing uncooperative customers, dealing with their unrealistic expectations and satisfying their requirements seem very difficult".	These quotes support the reported problems faced by participants whilst learning to deal with RE challenges.

The results of the second open question ("What are your suggestions for improving the RE course") have been analysed using a similar method and are presented below. Figure 4.5 illustrates the first three steps of the quantitative data and Table 4.3 shows the results of the next two steps. In the immersion phase, 64 different answers were identified. Subsequently, similar answers were grouped together into 18 categories in the categorisation phase. In the phenomenological reduction phase, three themes emerged, which were interpreted as improve teaching approaches, more practical work, and working on RE tools. In the triangulation phase, quotes from the answers to the questions were used to support the interpretation of themes.

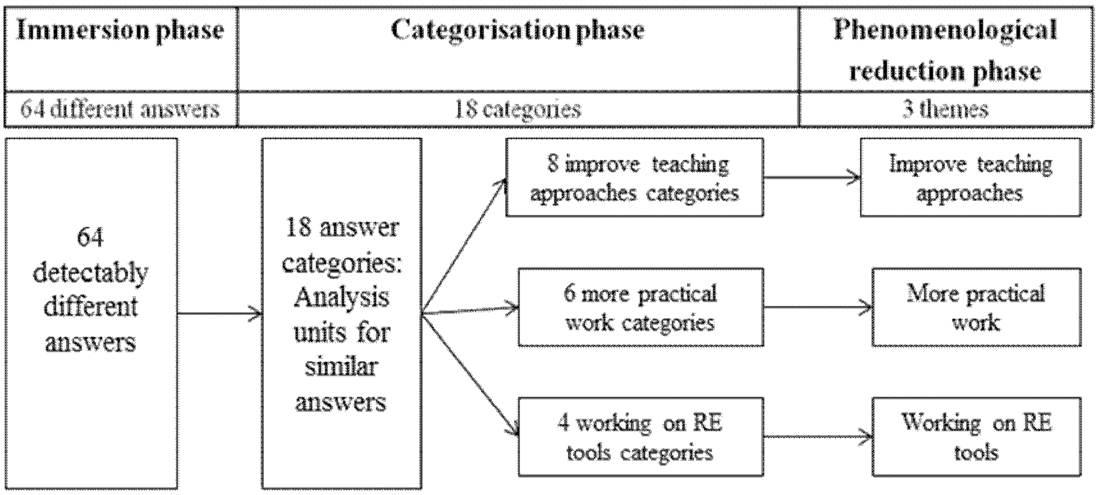


Figure 4.5: A diagram of the first three steps of the inductive analysis for the qualitative part of the questionnaire (suggestions for improving RE course taught in universities)

Table 4.3: Triangulation phase: Supporting quotes for the three themes of suggestions for improving the RE course taught in universities extracted from answers given by participants

Themes	Supporting quotes	Interpretation
Improve teaching approaches	"Provide detailed information and examples for each phase. Training should be provided so that students have sufficient RE knowledge". "Make learning session more interesting and interactive, more group activity/discussion". "Apply what has been taught to real life practice; simulate, for example, how real companies gather and analyse requirements". "The course is good enough, but there should be more attention on different tasks, e.g. creating SRS documents and initial problem structure, as well as validating customer requirements".	These quotes support suggestions provided by participants on improving current methods of teaching the RE course.

Table 4.3, continued

More practical work	“Make RE course more realistic, like meeting customers and stakeholders for requirements, not only learning about the theoretical side”. “Getting students to deal with real case scenarios instead of just listening to lectures” “Expose students to the industry and let them experience a real work environment” “Visits to organisations mainly working in requirements engineering field should be arranged for students. This would help them to understand this field better”. “Students should be provided with the facilities to work on real industry projects.	These quotes support suggestions provided by participants on how to include practical work in the RE course.
Working on RE tools	“Step-by-step guide on using RE tools should be given”. “Universities must provide the requirement engineering tools to use”. “Proper labs must be conducted to practically show real projects scenarios so to enhance student’s interest in this subject”. “Students should be taught to work using RE Tools”	These quotes support suggestions provided by participants on using RE tools in the course.

4.4.1.4 Discussion

The investigation results verified almost all of the problems presented in the integrated view. In addition, major problems faced by the students in REE have been extracted. The results of the second part of the questionnaire have led us to the five most difficult issues chosen by students from those presented to them, which are “working on RE tools”, “problem structuring and analysis”, “dealing with changing requirements”, “dealing with incomplete requirements” and “dealing with customers’ unrealistic expectations” (see figure 4.3). While the results and analysis of the open questions show that students were facing difficulties and need improvements in understanding RE concepts, working on RE activities (mainly in requirements elicitation, requirements analysis and requirements documentation), a lack of practical work, working on RE tools and RE challenges (see figure 4.4 and table 4.2).

From both of these results, it can be noted that:

- The problem of “Working on RE tools” is redundant and presented in both of the results, so it will be presented only once.
- The problems dealing with changing requirements, dealing with incomplete requirements and dealing with customers’ unrealistic expectations come under the category of “facing RE challenges” and will be automatically addressed if the problem of facing RE challenges is addressed. Therefore, these three problems are not considered as major problems.

This left us with only six REE problems which are problem structuring and analysis, working on RE tools, understanding RE concepts, working on RE activities, lack of practical work and facing RE challenges. These problems can be considered as major REE problems faced by the students.

4.4.2 Lecturers' investigation results and analysis

In the lecturers' questionnaire, all three parts contain quantitative (closed questions) as well as qualitative (open questions) data. The data analysis and results of the three parts are presented in the following sections.

4.4.2.1 RE elements and challenges

Based on the results, it was found that the lecturers surveyed teach RE courses through lectures (reported by 100% of the respondents), supplemented with other approaches such as labs (72%), presentations (72%) and group discussions (78%). The lecturers' responses to the first part, which is about RE elements and challenges, are presented in Figure 4.6, which shows that RE elements are mostly emphasized while teaching the RE course. However in the case of RE challenges, a few emphasized it in the class while others either emphasized it less or not at all. In the open part of this question, the lecturers were asked to give a reason if any of the RE elements or challenges was given less emphasis or was not emphasized at all in the class. The reasons given for not emphasizing any of the RE elements were either they had "insufficient time to cover the topics in one semester" (22%) or "the topics were not covered in the syllabus" (5%). Meanwhile, the reasons given as to why RE challenges are seldom taught explicitly are because majority of the lecturers feel that RE challenges are already covered in the main RE activities and are not included explicitly in the syllabus. Therefore, they can be considered as side topics. According to a few of them, RE challenges such as resolving conflicts, defining scope, and facilitating decisions are more towards project management. The majority of lecturers also set class projects for students to let them practice RE activities.

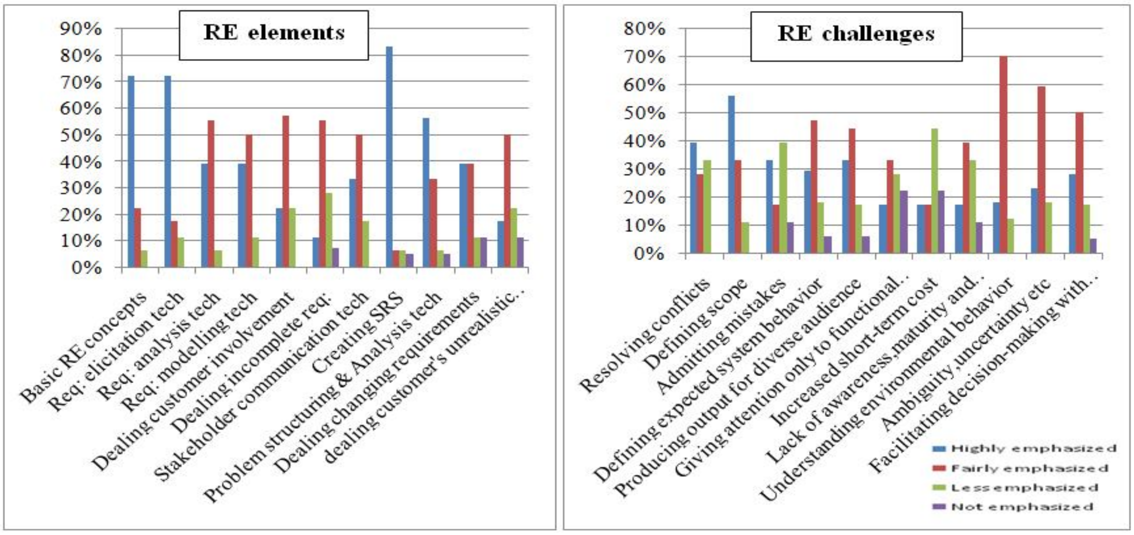


Figure 4.6: Lecturers' responses to RE elements and challenges

4.4.2.2 RE issues

The lecturers' responses to the second part of the questionnaire which is about RE issues is shown in Figure 4.7. The results suggest that dealing with changing requirements is the most difficult to teach and is reported by 50% of respondents, followed by problem structuring and analysis (40%), dealing with RE challenges (40%) and producing good quality requirements specification (39%), while the other issues are less difficult to teach. In the open part of this question, the respondents were asked the reason why they found these issues difficult to teach. Only a few of them elaborated on their choices. The most frequently cited response was "It was difficult to set up situations for students to practice" (22%), followed by the other two themes/reasons (Table 4.4).

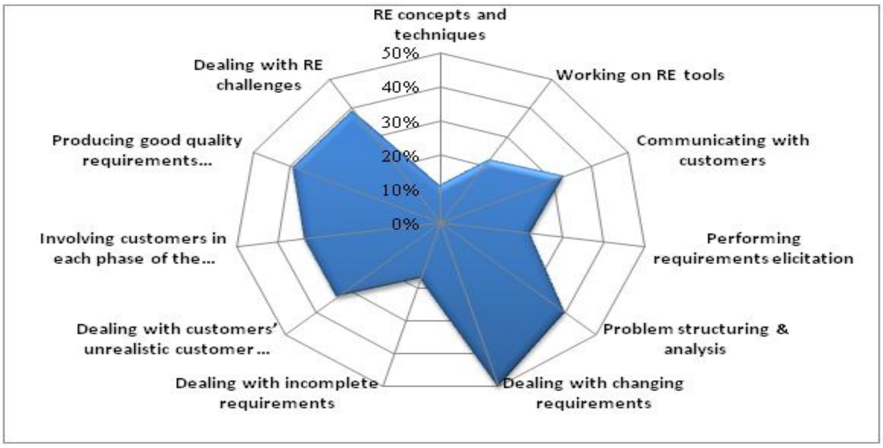


Figure 4.7: The lecturers' responses on RE issues

Table 4.4: The lecturers' elaborations on the selection of RE issues

Theme	**F (%)**
Hard to setup situations for students to practice RE issues	22
Communication with customer needs a lot of resources	17
Working with RE tools become difficult because students take it lightly	6

F = Frequency

4.4.2.3 REE problems and suggestions

In the third part of the questionnaire, the lecturers were asked about the problems they faced while teaching RE in order gain more insights into REE problems and to gather suggestions for improving RE courses, by means of two open questions. The data was analysed using the constant comparison method (Strauss & Corbin, 1998) in which all detectably different qualitative answers were recognised and divided into categories. The themes that emerged from the categories as a result are presented in Table 4.5 and 4.6.

Table 4.5: Problems faced while teaching RE

Theme	F (%)
Teaching and getting students to grasp RE is difficult because of the theoretical nature of RE	38
Providing real world experience of RE to students in the classrooms is difficult	17
Finding good case studies to be used in teaching RE	15
Finding proper RE tools	15
Teaching students to write requirements specifications	15

F = Frequency

Table 4.6: Suggestions for improving RE course

Theme	F (%)
Should provide students with practical experience by • Giving them practical examples • Using real projects • Let them experience RE activities • Providing lab exercises • Involving them in industrial projects	56
Efficient RE tools should be made available	17
Proper course outline should be prepared, keeping industrial needs in mind	11
Concentrate on problem based learning	11

F = Frequency

4.4.2.4 Discussion

The data analysis results verified the problems presented in the integrated view and also show the major problems faced by lecturers in REE. According to the results:

- The issues that lecturers find most difficult to teach from the ones presented to them are dealing with changing requirements, problem structuring and analysis, dealing with RE challenges and producing good quality requirements specifications (see figure 4.7).

- While the problems extracted from open questions (stated by the lecturers) are providing organizational experience in REE, teaching skills in order to produce good quality requirements specifications and working on RE tools (see table 4.5).

From both of these results, the following observations can be taken:

- The problem of producing good quality requirements specification is redundant so it will be presented once.
- The problems dealing with RE challenges and dealing with changing requirements comes under the category of “providing organizational experience in REE” and can be automatically addressed if the problem of providing organizational experience is addressed, therefore these two problems are not considered as major problems.

This leaves us with four problems which are problem structuring and analysis, producing good quality requirements specifications, providing organizational experience in REE and working on RE tools. These problems can be considered as the major REE problems faced by the lecturers.

4.4.3 Limitations

To ensure representative coverage, our subjects included users and developers of various levels of experiences, qualifications and backgrounds.

Clearly, an important limitation of the student’s study involves the small sample size (89 students), the relatively homogenous population (undergraduate software engineering students from two Asian countries) and the short duration (4 weeks). In addition to this, the pre-test was not performed prior to study. This severely limits the external validity of this study.

Fortunately, the goal of this study is to verify REE problems presented in the literature. Because of the factors that the RE course is being taught using standard topics and due to the nature of RE, students usually face problems in learning RE concepts. Therefore, it is expected that a replication of this study in a different site and/or with different size teams shall generate the same results. However, all the lecturers participated in the study are either involved in teaching RE course or in doing research in RE in different countries. Therefore the results can be generalized.

4.5 REE research gaps

The REE research gaps are identified by analysing the major problems faced by the students and lecturers with reference to the problems presented in the integrated view. In the integrated view, the problems that were only reported but not investigated by researchers present the areas that need further investigation. There are five such problems in the integrated view (see Figure 3.4). Of those five problems, “the need for students to understand the importance of RE” is a problem that requires didactic skills, while the problem “lack of RE challenges” is a general problem whereby it is assumed that students can learn to deal with the challenges if the rest of the RE practices problems are addressed. This leaves us with the remainder three problems which are teaching students to analyse and structure real world problems from customers (or teaching problem structuring and analysis), dealing with unrealistic customers’ expectations, and the lack of customers’ involvement in projects. These three problems can be seen as the results of the literature investigation, and are combined with the results of the students and lecturers investigations in order to identify the REE research gaps. Table 4.7 shows the results of these types of investigations.

Table 4.7: The results of the REE problems investigations

1)	Major problems from integrated view (Literature investigation)	Teaching problem structuring and analysis Dealing with unrealistic customers' expectations Lack of customers' involvement in projects
2)	Major problems faced by students (Students' investigation)	Problem structuring and analysis Working on RE tools Understanding RE concepts Working on RE activities Lack of practical work Facing RE challenges
3)	Major problems faced by lecturers (Lecturers' investigations)	Problem structuring and analysis Producing good quality requirements specifications Providing organizational experience in REE Working on RE tools

By combining the results of the three investigations and removing the redundancies and dependent problems, the REE research gaps are identified. For example, the problem of "teaching problem structuring and analysis" appears in all three results while the problem "working on RE tools" appears in two out of the three results. Therefore, these problems are reported once to remove redundancy. Meanwhile, the two problems of "dealing with unrealistic customers' expectations" and "lack of customers' involvement in projects" can be seen as dependent on the problem of "facing RE challenges" and therefore are not included as research gaps. Moreover, the problem of "producing good quality requirements specifications" can be addressed if the problem of "working on RE activities" is addressed because the requirements specification is produced as an output of the requirements documentation which is among the major RE activities. Thus, this problem is not included as research gaps as well.

The problem “lack of practical work” is dependent on “providing organizational experience in REE” and can be addressed if the students are provided with organisational experience while teaching RE; therefore this problem is also not included as research gaps. This analysis resulted in the identification of six research gaps in REE as presented in Figure 4.8.

Figure 4.8: REE research gaps

4.6 Formulating the research focus

In order to formulate the research area that contributes towards addressing the REE problems, we need to look back on the researcher’s efforts in addressing the major REE problems. It was observed that the researchers have worked and tried to address many of these problems, such as the problem of providing organizational experience in REE which was investigated by (Regev et al., 2009), whereby they used experiential learning approaches using low-tech social simulations in order to provide practical experience to students in REE. Meanwhile, the problem of working on RE tools was investigated by (Rosca, 2000) where he emphasized the need to introduce students to the RE processes, methods and tools and used role-playing, lectures and laboratory work as pedagogical approaches to investigate the problem.

The problem of understanding RE concepts was investigated by (Al-Ani & Yusop, 2004), whereby they also used role-playing and peer assessment as pedagogical approaches in order to introduce students to the foundations of RE. The problem of working on RE activities has been investigated by many researchers such as (Connor et al., 2009), (Zowghi, 2009), (Barnes et al., 2008) and (Beatty & Agouridas, 2007), where they proposed different strategies to address this problem. The problem of facing RE challenges was reported as well as investigated by many researchers such as (Connor et al., 2009), (Zowghi, 2009), (Barnes et al., 2008) and (Beatty & Agouridas, 2007).

Although the researchers proposed solutions to address these problems, according to the investigations' results, the students and lecturers still reported encountering these problems. The possible reasons can be the lack of awareness of the academicians on current research, or the research results are not being practically adopted by the academicians, or some of the proposed solutions may be proven ineffective when adopted into practice. Therefore, further REE research is needed to address these problems and to adopt the research results into academics.

Of the major REE problems presented in Figure 4.8, the only problem that has yet to be investigated is the teaching of problem structuring and analysis. Students are facing problems in understanding this issue because it is not explicitly taught in RE course. Although reported by many researchers in the literature such as (Barnes et al., 2008), (Beatty & Agouridas, 2007) and (Gibson, 2000), the teaching of problem structuring and analysis has yet to be investigated, therefore it appears to require more attention within the REE research. Problem structuring and analysis is a process that encompasses learning about the problem to be solved, understanding the needs of potential users, understanding all the constraints on the solution and the organization, and structuring this large volume of information to make it more understandable (Sommerville & Kotonya, 1998).

While teaching RE course in universities, the aspect of problem structuring and analysis is paid very little attention (Rogers, 1999). In classrooms, the emphasis is to teach students problems that are well structured and well understood, and requirements that are well captured and completed. Therefore, this problem was selected to be investigated further in this research.

4.6.1 Verification from lecturers

In order to verify that the selected research area will optimally contribute towards addressing the REE problems and that the research pursued is relevant to the current needs of academicians and researchers involved in this area, it was felt necessary to ask lecturers and researchers about their opinions, recommendations and feedbacks on selecting and investigating this problem. This was accomplished through questionnaire survey from the same lecturers who participated in the previous study.

The questionnaire includes two open questions (see Appendix C). In the first question, the lecturers were asked to give their comments, feedback and suggestions regarding the selected research focus. The most cited comment is "Problem structuring and analysis is a crucial aspect of RE" (50%), while the other two themes describe this aspect as very critical but difficult to teach in the classrooms, and that it can be taught by using various approaches (see Table 4.8).

Table 4.8: Lecturer's comments and feedbacks

Theme	**F (%)**
It is a crucial aspect of RE	50
Critical and difficult to teach	20
Need various approaches	18

F = Frequency

Along with these comments, a few suggestions by lecturers are

- Put a huge focus on understanding the objectives once the problems are understood.

- Make use of an appropriate and elaborate case study, where the problem structure is nontrivial.
- It would need interaction with a customer. You cannot structure their problems without dialogue.
- Understand the approaches and techniques for optimizing, understanding, analysing and structuring initial requirements in the literature and try to propose new ones or enhance/refine/extend the existing ones to make it simple and effective for students' understanding.

The second open question asked lecturers whether they believe that this is an area that requires research attention. Almost all of the lecturers responded positively by stating that this is a potential area of research and that this problem needs to be addressed.

Thus, it can be argued from the survey results that almost all of the respondents i.e. the lecturers surveyed, encouraged this problem to be investigated further by stating that this is a critical area that is not often paid much attention to. Also they argued that it is very difficult to make students understand this aspect of RE. This research therefore, investigates the problem of teaching problem structuring and analysis in RE in order to find an effective way to address this problem.

4.7 Summary

This chapter presented the results of two investigations performed on students and lecturers. The investigation results verified the problems presented in the integrated view. The REE research gaps were identified through an analysis of the investigations results. Further analysis of the research gaps have resulted in the selection of one problem area as the research focus for this study which is the problem of teaching problem structuring and analysis in RE. The selection of the research focus was supplemented by performing another investigation on the same lecturers who have taught RE course. The investigation results showed that the lecturers consider this problem as a crucial aspect of RE and highly recommend this problem to be investigated.

Chapter 5- Related work on the teaching of problem structuring and analysis

5.1 Introduction

This chapter discusses in detail the need for teaching students problem structuring and analysis in RE course and gives an overview of problem structuring and analysis process in RE. It then presents existing problem structuring and analysis methods, and identifies the research gap that acts as a motivation to develop a new method for the process of teaching problem structuring and analysis.

5.2 The need for teaching problem structuring and analysis in RE

Traditionally, research in software development has focused on solutions, i.e. on programs and on various abstractions that may be useful in designing and writing program texts. Researchers have paid little attention to the problems that those programs are intended to solve. This solution-oriented approach may work well in a field where the problems are well-known and have been thoroughly described, classified and investigated, and where innovation lies only in devising new solutions to old problems. But software development is not such a field (Jackson, 1999). The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements. No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later (Brooks, 1987). One of the problems with current practice in software development is that customer requirements are often not well captured, understood, analysed and structured. This problem often leads to a mismatch between what the customer needs and what the software developer understands the customer needs (Li, 2008).

Normally in our rush to develop techniques for solving problems, we accept initial formulations as givens, fail to fully explore the boundaries of the problem, and shorten the creative process by making a premature evaluation. Finding opportunities rather than simply solving the problem-as-given is the real challenge (Volkema & Evans, 1995). Because if too little work is done in understanding stakeholder requirements, there is a great risk that the product will not meet the stakeholder expectations, thus requiring effort to rework the product to bring it into compliance with stakeholder needs (McPhee & Eberlein, 2000). Therefore, extracting and understanding customers' problem and organizing it into a set of user requirements is the most important function that a software requirements engineer must perform, and these requirements can then serve as a basis for software design. Almost inevitably, this involves considerable widening and recasting of the original problem statement. In some cases, it may result in a whole set of problems being defined (Woolley & Pidd, 1981). However, this aspect is not paid much attention by requirements engineers while doing software projects, academics while teaching RE to students in universities (Rogers, 1999), as well as by researchers while doing RE research.

To ensure software engineers have the essential skills to perform the problem structuring and analysis process, it is recommended that it should be taught in RE course to software engineering students in universities. The importance of this aspect should be taken into account due to the situation of the labour markets that the students will be joining after graduating. When software engineering students first wrestle with real large-scale software problems, either in a classroom or as part of a software project, they face an impasse. On one hand, there is the problem statement, in all its ambiguity, incompleteness, and lack of structure.

On the other hand, there is the promise of organized design, through concrete, familiar notions of architecture, objects, and code. The newly graduated students are great in solving well-structured problems, but they are unable to structure problems from real messes and get the right problem definition (Rogers, 1999). On one hand, it is convenient in education, both in specification and assessment, to provide fairly well-structured problems, and many instructors view using such problems as a way to manage the learning process. However, real-world problems are typically ill-structured and it can be argued that using only well-structured problems as learning examples does not prepare students for the problems that they will encounter in their professional life. Preparing students for dealing with ill-structured or open ended problems is an educational challenge involving critical thinking skills, which most instructors and curriculum designers view as an important goal of the learning process (Daniels, Carbone, Hauer, & Moore, 2007). Unfortunately this problem structuring and analysis aspect of RE have traditionally been less investigated by RE researchers as well as RE academicians. The few recent proposals such as (Daniels et al., 2007), (Smith & Gotel, 2007), (Beatty & Agouridas, 2007) and (Barnes et al., 2008) and (Gibson, 2000) reported the problem of teaching problem structuring and analysis in REE but the problem has not been fully investigated yet.

An integrated view of REE problems by (Memon, Salim, & Ahmad, 2013) also presented the problem of teaching problem structuring and analysis as an important problem that requires more attention in REE research. The analysis of REE problems presented in the integrated view was performed in another study (Memon, Salim, & Ahmad, 2012) and the results showed that this problem has been less investigated by researchers, therefore it can be considered as a major gap in REE research that needs to be filled.

While teaching RE course, problem structuring and analysis is not emphasized explicitly but usually thought to be covered using requirements analysis methods. But the requirements analysis methods focus more on requirements generation and are meant for experienced requirements engineers working in industry. For students who have usually no industrial experience, it is difficult for them to understand these methods and perform problem structuring and analysis using these methods. Therefore, a light weight method is needed for students that can enable them to learn and apply the process of problem structuring and analysis.

5.3 The problem structuring and analysis process

This section defines problem structuring and analysis, and shows the scope of this research by presenting the role of the problem structuring and analysis process in RE.

5.3.1 The problem and its characteristics

The literature defines the term "problem" in several ways; a few of them are presented here. The problem can be defined as, "a situation where someone wanted something to be different from how it is and is not quite sure how to go about in making it so" (Badal, 2006). Another definition presented by (Evans, 1991) is "The problem is a gap between the present and some desired state of affairs". While (Agre, 1982) defined the problem as, "an undesirable situation that is significant to and may be solvable by some agent, although probably with some difficulty."

The problems from customers are often ill-structured and different researchers have used different terms to describe these ill-structured problems, such as (Rittel & Webber, 1973) called these situations wicked problems; while (Schön, 1987) called these types of problems swamps or swampy situations, whereas (Barnes et al., 2008) and (Ackoff, 1979) termed this type of problem a "mess".

For understanding and explaining what a problematic situation is, it is interesting to refer to the explanation of (Vidal, 2002). He characterized a problematic situation or mess as follows:

- Highly complex situations, due to many factors, many actors, lack of structure, many interrelated and objective and subjective aspects.
- Lack of internal transparency, due to many uncertainties about the reactions of the actors, many interrelated communication channels, and internal power relationships.
- Several conflicting goals, due to the lack of agreement about the visions and mission of the organization.
- A whole network of interrelated problems of change in the organization.
- Dynamic situation, due to a permanent interplay between the organization and the environment.
- Lack of technological and methodological expertise in the organization. (Vidal, 2002)

5.3.2 Definition of problem structuring and analysis

Problem structuring and analysis can be seen as a combination of two terms, “problem structuring” and “problem analysis”. The term “problem structuring” is mostly used in MS/OR (Management Sciences/ Operational research) literature, and is defined as “the process of formulating the present set of conditions, symptoms, causes and triggering events into a problem or set of problems sufficiently well specified so that the risk of using analytic procedures to solve the wrong problem has been minimized” (Schwenk & Thomas, 1983). The term “problem analysis”, on the other hand, is usually used in requirements engineering and is defined as “the process of understanding the customers’ real problem, and then translating that understanding into a set of real needs (Sidky, Sud, Bhatia, & Arthur, 2002).

The combination of both of these terms that is “problem structuring and analysis” was first used together by (Jackson, 1999) and (Wallace, Wang, & Bluth, 2006) to cover the process that encompasses learning about the problem to be solved, understanding the needs of potential users, and organizing and structuring this information into a set of requirements to make it more understandable (Sommerville & Kotonya, 1998). It can therefore be inferred from the above definitions that problem structuring and analysis consists of two main processes, i.e. first, identifying and understanding the real problem and causes of the problem, and second, decomposing, organizing and refining the problem into a set of requirements for a desired system.

5.3.3 Problem structuring and analysis in RE

RE consists of six major activities which are requirements elicitation and analysis, requirements analysis, requirements modelling, requirements documentation, requirements verification and validation, and requirements management. Requirements elicitation and analysis is the process by which a customer’s needs are understood and documented to arrive at a definition of software requirements (Thayer et al., 1997). The goal is to understand the customer's business context and constraints, the functions that the product must perform, the performance levels it must adhere to, and the external systems it must be compatible with. Techniques used to obtain this understanding include customer interviews, use cases, brainstorming etc. (Melonfire, 2007). However, this process is considered ill-defined (Sidky et al., 2002) because it does not cover all aspects of analysis but is performed using only a few analysis techniques which is insufficient.

The requirements analysis process requires complete understanding and partitioning of the problem as well as organizing the knowledge into a set of requirements that can be used as an input to the requirements modelling activity. Therefore, the problem structuring and analysis process should be performed as part of requirements elicitation and analysis. In Figure 5.1 we have placed problem structuring and analysis in perspective to the major requirements engineering activities.

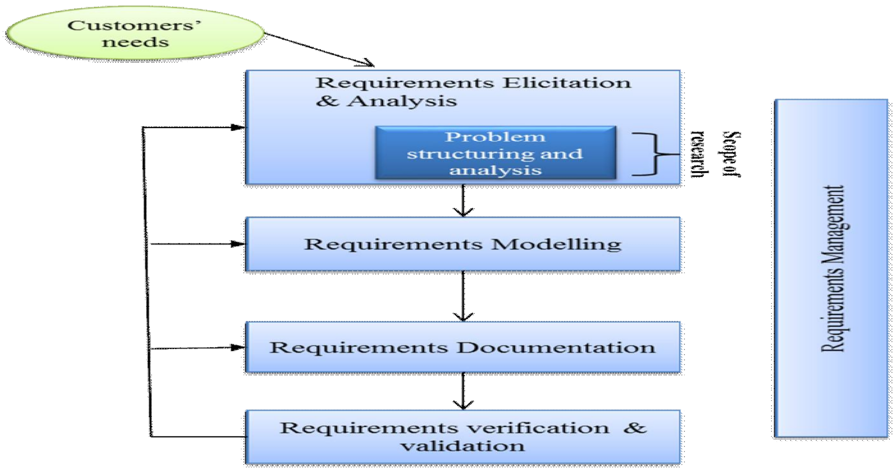


Figure 5.1: Problem structuring and analysis in perspective to RE

The process uses initial customers' needs as an input, on which the analysis is performed. While performing the problem structuring and analysis process, the customers' requirements should be elicited further using requirements elicitation techniques and sent for analysis and structuring. Along with problem structuring and analysis, further analysis can also be performed on the requirements produced, which can then be used as an input to the requirements modelling activity. The scope of this research is to cover the problem structuring and analysis part of the requirements analysis.

5.4 Existing methods for problem structuring and analysis

In order to formulate the system requirements, there is no ideal requirements method used. Instead, a number of methods that use a variety of techniques is used to perform the requirements process (Kotonya & Sommerville, 1998). The methods and approaches that claim the title of 'problem analysis', on closer inspection, usually prove to deal entirely with putative or outline solutions. The problem to be solved is neither stated in full detail nor explicitly analysed; the reader must infer the problem from its solution (Jackson, 1999). Several methods have been proposed to be used at the early stages of software development. We shall now outline those methods which are particularly close to our subject matter to see whether they can help us in performing the problem structuring and analysis process.

In this section, we focus on three aspects, 1) problem structuring methods, 2) requirements analysis methods and 3) the teaching of problem structuring and analysis in RE.

5.4.1 Problem structuring methods (PSMs)

PSMs are a broad group of problem-handling approaches whose purpose is to assist in structuring problems rather than directly solving them (Rosenhead, 1996). Most of the PSMs are proposed in the field of MS/OR.

(Pidd & Woolley, 1980) categorizes PSMs according to four streams of thoughts which are check list stream, definition stream, science/research stream and people stream. But they claim that each of these streams appears to rest on a very constrained view and none of them provides the key to problem structuring. That is why people in practice probably do not use any of these approaches (Pidd & Woolley, 1980). The exploration stream aims to overcome the deficiencies of the prior four streams, and incorporated personal observations and experiences. It is characterized by four fundamental aspects which are informality, hierarchy, continuance and inclusiveness (Pidd, 1988).

Strategic Options Development and Analysis (SODA) is another method for working on complex problems (Eden, 2004). It relies on the concept of cognitive mapping to represent a person's thinking on an issue in the form of a directed graph consisting of nodes (ideas) and arcs (connections between ideas) (Ellspermann, Evans, & Basadur, 2007). However in SODA, the work tends to be more "quick and dirty" than thorough and complete (Eden, 2004).

Soft Systems Methodology (SSM) was developed by Peter Checkland (Checkland & Scholes, 1999) at Lancaster University for the express purpose of dealing with problems that are ill structured, messy, changing and poorly defined. (Badal, 2006). SSM analyses the problem from two perspectives which are real world thinking and system thinking. SSM acts well as a structuring frame of the studied problem (Sorensen & Valqui Vidal, 2008) and emphasizes on problem identification, problem structuring, and problem resolution rather than on problem solution. The methodology has seven steps, 1) Finding out the problem situation, 2) Expressing the problem situation through pictures, 3) Selecting how to view the situation and producing root definitions, 4) Building conceptual models of what the system must do for each root definitions, 5) Comparison of the conceptual models with the real world, 6) Identifying feasible and desirable changes, and 7) Recommendations for taking action to improve the problem situation. (Badal, 2006)

Problem frames is an approach to problem analysis and structuring in RE that characterizes classes of problems that commonly occur as sub problems of larger and realistic problems with an intention to analyse realistic problems by decomposing them into constituent sub problems (Jackson, 1999). Indeed, even if a problem decomposition structure succeeds in breaking down the complexity of a problem and allows a suitable solution to be developed, such a structure may not be easy to evolve as the original problem and its requirements change (Barroca, Fiadeiro, Jackson, Laney, & Nuseibeh, 2004).

The information on these PSMs is summarized in Table 5.1 including their background, focus, process and so on.

Table 5.1: An overview of the problem structuring methods

Characteristics of	**Four streams**	**Exploration stream**	**SODA**	**SSM**	**Problem frames**
Background	Operation Research / Management Sciences	Operation Research / Management Sciences	Psychology/ Social psychology	Systems Engineering	Software Engineering
Focus	Structuring of messy problem situation	Structuring of messy problem situation	Support in perception and structuring of a messy problem situation	Structuring of a messy problem situation	Problem analysis and structuring

Table 5.1, continued

Process	Identifying the elements of problems by asking questions, gathering data, observations in terms of objectives, people, and different measures	Learning process where an analyst strives to comprehend the full complexity of the issues and considers carefully how to manage this complexity	Learning process where dialectic thinking comes from analysing individual perceptions and these are gathered into an aggregated model	Learning process where individual world views are described and systematized	Analyses realistic problems by decomposing them into constituent sub problems that correspond to known problem frames
Organisation	Individual interviews and workshops with interactive participation & observations.	An analyst learns about the problem and its context by exploration.	Individual interviews and workshops	Workshops with interactive participation	Analysis and decomposition of complex problems by a specifier.
Technology	Checklists, and decision models	An open-ended descriptive model	Cognitive maps	Systematic & organized thinking	Context/ Problem diagram
Consultant function	Analyst	Analyst with divergent skills	Facilitator and analyst	Facilitator and expert in methodology	Analyst

Each PSM covers an initial analysis of problem and offers a way of representing the situation that will enable participants to clarify their predicament, converge on a potentially actionable mutual problem or issue within it, and agree on commitments that will at least partially resolve it (Mingers & Rosenhead, 2004). Two main complications of these PSMs are in practice they are not as easy to apply as their creators affirm, and in some of them, the method is too strict and no creativity can be introduced (Badal, 2006). The problem frames although aimed at problem structuring and analysis process, focuses on decomposing the problem into solution pieces that are known to already exist. Hence, it can be used to extract those components that have existing solution templates from the original problem. This is an example of a "solution-oriented" approach to problem decomposition (Sidky et al., 2002).

5.4.2 Requirements Analysis methods

In the field of RE, a number of requirements analysis methods have been proposed that can cover the problem structuring and analysis process. For example, goal-oriented requirements engineering is an approach to requirements analysis that is concerned with the use of goals (an objective the system under consideration should achieve (Van Lamsweerde, 2001)) for eliciting, elaborating, structuring, specifying, analysing, negotiating, documenting, and modifying requirements. Goal-based methods stress the need to characterize, categorize, decompose and structure goals as requirements, but usually fail to offer strategies to identify goals, taking it for granted that the goals have already been documented (Anton, 1996).

Scenario driven requirements analysis methods on the other hand, analyse the problem using scenarios (sequence of events that occurs during one particular execution of a system). They are used to aid in identifying requirements before and as part of building a system (Wang, Hufnagel, Hsia, & Yang, 1992).

Structured Analysis and Design Technique (SADT) consists of techniques for performing system analysis and design (Ross & Schoman Jr, 1977). The only function of SA is to bind up, structure, and communicate units of thought expressed in any other chosen language (Ross, 1977). But structured analysis and design methods are not considered very useful for the early stages of requirements analysis because they are based on hard and inflexible models of system such as entity relationship models, data flow models. (Kotonya & Sommerville, 1998).

Finally, Viewpoint Analysis is based on the acknowledgement that software requirements can be elicited from different viewpoints. It shows that comparing different perceptions about a problem helps in the understanding of the problem being addressed (Leite, 1989).

The information on these requirements analysis methods is summarized in Table 5.2 including their background, purpose, process and so on.

The requirements analysis methods usually focus on the later stages of analysis (i.e. producing requirements from the user needs as given), hence they are not considered suitable for an initial analysis of the customers’ problem (Kotonya & Sommerville, 1998).

Table 5.2: An overview of the requirements analysis methods

Characteristics of	**Goal-oriented requirements engineering methods**	**Scenario Driven requirements Analysis methods**	**Structured Analysis and design techniques**	**Viewpoint Analysis methods**
Background	Requirements Engineering	Requirements Engineering	System Engineering	Requirements Engineering
Purpose	Eliciting, structuring, analysing, negotiating, documenting, and modifying requirements	Eliciting and analysing the requirements	System Analysis and design	Eliciting, analysing and validating the requirements
Process	Describe system objectives and carry out process to meet those objectives	Identifying and understanding requirements by describing sequences of events	Applying system analysis and design techniques in requirements definition and system development	Understanding of the problem by comparing different perceptions about the problem
Technique	Goals	Scenarios	Structured analysis models	Viewpoints
Consultant function	Requirements analyst	Requirements analyst	System engineer	Requirements analyst

5.4.3 The teaching of problem structuring and analysis

Requirements engineering methods have been developed to assist in eliciting, analysing, modelling, specifying and managing requirements. Teaching these methods to university students should help the students to develop a more complete, consistent, and testable requirements. The teaching of system requirements analysis in a university setting is difficult because of the breadth of knowledge and experience that is required. Understanding the big picture that the technical nuts and bolts fit into is difficult for students, but this knowledge is required before a beginning engineer's knowledge can become operational (Gonzales & White).

The problem structuring and analysis process is being taught using available requirements analysis methods to software engineering students in the RE course in universities. However, the focus of the requirements analysis methods is more towards generating requirements rather than on the problem to be solved. Therefore it is difficult to call students' attention to the benefits of focusing on the problem to be solved. Yet without a means to discuss and inquire about problems in a precise and structured way, students can hardly be faulted for skipping problem analysis and leaping straight to the comforting rigor of design (Wallace et al., 2006).

Experienced requirements engineers can use the existing requirements analysis methods to understand and structure problems and extract requirements based on intuition and experience (Sidky et al., 2002). But for students with no industrial experience, it is difficult to learn to perform problem structuring and analysis using these methods.

5.5 Motivation for method

Among the problems presented in the integrated view, the problem of teaching problem structuring and analysis were found to be less investigated by researchers. Even though the literature survey on this problem was performed, it was still difficult to identify a teaching method that can help students in learning and performing the problem structuring and analysis process. The objective of this study is therefore to find an approach suitable for students that can enable them to understand and apply this process.

Problem structuring and analysis process should cover two stages. First, it should enable a person to have a good, general understanding of a problem and its causes; stakeholders' collective input and judgment will determine the nature of a system and the initial requirements of the system, which are abstract and normally of high level. Second, it should enable a person to partition a problem into high level requirements and then decompose these high level requirements into features and then function level requirements.

Existing problem structuring methods normally work only on the initial understanding of a problem and its causes, but the process does not continue further to generation of requirements. Requirements analysis methods are usually used to analyse production of requirements, but the process does not concentrate on initial understanding of the problem. Therefore, this one piece of puzzle that enables a person to generate customer needs (requirements) from elements of a decomposed problem set particularly needs additional exploration and tool/method development. Currently this activity relies mostly on intuition and experience (Sidky et al., 2002).

For novice requirements engineers/students, it is very difficult to perform both stages of problem structuring and analysis process and to move from the stage of initial understanding of a problem to the stage of generation of requirements using existing methods. Therefore, they need to learn problem structuring and analysis process using an easy-to-understand and easy-to-apply method that covers both stages of the process; the lightweight method can help to bridge the gap between the two stages.

In addition to the problem structuring and requirements engineering methods presented in sections 5.4.1 and 5.4.2, a framework that can be used for performing problem structuring in requirements engineering is i* framework. i* framework is a modelling language suitable for an early phase of system modelling in order to understand the problem domain. i* modelling language allows to model both as-is and to-be situations. It covers both actor-oriented and agent-oriented modelling. (Franch, 2012) However this approach originally developed for modelling and reasoning about organizational environments and their information systems composed of heterogeneous actors with different, often competing, goals that depend on each other to undertake their tasks and achieve these goals. Also, i* models answer the question WHO and WHY, not what. (Franch, 2012) Therefore, the framework was not considered suitable to be used for teaching students problem structuring and analysis process.

In our opinion, SSM is the only problem structuring method that can be proved effective and suitable for the purpose of this study; as the method helps in understanding a problem, organisational context of a problem and constraints of solution to a problem. SSM is not specifically designed as a technique to elicit and analyse requirements for computer-based systems, but it provides a means to understand abstract system requirements through analysis of organisational context, the problem to be solved and existing systems in place.

SSM is very useful for early-stage analysis where application domain, the problem and organisational requirements must be understood (Kotonya & Sommerville, 1998).

Therefore, in this study, we adopt the ideas of SSM to cover the two stages of problem structuring and analysis; we will look at the problem from two perspectives, which are real-world thinking and system (or software in our case) thinking. The first perspective is the view of the world we are trying to understand and how things are done (What it is); the second perspective is the view of how things should work in the future, or the product we plan to build (What it will be) (Robertson & Robertson, 2012). However the ideas of SSM need to be translated into the form of lightweight RE method, so that it is easier for students to understand and perform the problem structuring and analysis. The development of L-Soft method is, therefore, prompted by the need for a lightweight problem structuring and analysis method for students in the field.

5.6 Summary

This chapter presented the problem of teaching problem structuring and analysis as reported by researchers in the literature and described its role in RE. The existing problem structuring and requirements analysis methods have been presented, and the research gap has been shown that motivated us to propose a method. The next chapter presents the proposed method.

Chapter 6- L-Soft: A light weight method for teaching problem structuring and analysis

6.1 Introduction

This chapter presents the method proposed for teaching problem structuring and analysis in RE course. The objective of the method is to provide students with an easy to understand and apply approach for understanding and structuring customers' problem into a set of requirements. The method is aimed at undergraduate software engineering students undertaking RE course. This chapter first discusses the highlights of the method and then presents in detail the L-Soft method.

6.2 Highlights of the L-Soft method

The proposed method is intended for teaching students to perform the process of problem structuring and analysis in RE course. In order to cover both aspects of problem structuring and analysis, from problem understanding up to the generation of requirements, and in a way that enables the students to understand and apply the process, a lightweight method for performing the problem structuring and analysis process has been proposed to be taught to the students during RE course. The method is referred as L-Soft to portray that it has transformed the idea of Soft System Methodology (SSM) into a light-weight RE method. In this section, a brief overview of the L-Soft method is presented by its characteristics.

6.2.1 Adoption of ideas of SSM into L-Soft method

To understand and structure the problem, SSM initially understands the problem using real-world thinking and then structure it using system thinking. To cover the two stages of problem structuring and analysis; the concept of looking at the problem from two perspectives, which are real-world thinking and system (or software in our case) thinking is borrowed from SSM.

The first perspective is the view of the world we are trying to understand and how things are done (What it is); the second perspective is the view of how things should work in the future, or the product we plan to build (What it will be) (Robertson & Robertson, 2012). The two perspectives are customised and used as two phases of the method that are problem space and software world. However, because our method is intended to be used in RE, the steps of the method are different from SSM (which normally been used in MS/OR). Also, the constructs used to perform steps of the method are mostly taken from RE to make the method easier for students to understand.

6.2.2 The glossary of terms used in the method

Several terms used in the L-Soft method are defined in the way that they are interpreted in the method.

- Problem: The difference between things as perceived and things as desired (Gause & Weinberg, 1990).
- Causes of problem: The reasons behind the existence of problems.
- Existing work process: Describes how the work is being done currently in the organization.
- Refined work process: The set of steps that needs to be carried out in order to achieve one or more functional goals.
- Goal: High level objectives of the business, organization or system (Anton, 1996) or something that some stakeholder hopes to achieve in the future. (Rolland, Souveyet, & Achour, 1998)
- Scenario: The sequence of steps performed in order to perform a work process.
- Feature: Represents the requirements at higher level that the product should support and can be decomposed into one or more lower level requirements.
- Functional requirements: The actions that the desired product shall perform.
- Non-functional requirements: The qualities that the desired product should have.

- Analysis checklist: List of questions through which each requirement is assessed to ensure acceptable quality.

6.2.3 Two viewpoints to cover both aspects of problem structuring and analysis

In requirements analysis, it is considered easier for analyst to look at an aspect of the world from a number of different viewpoints (Robertson & Robertson, 2012). L-Soft method analyses and structures a problem from two perspectives which are the problem space and the software world. In the problem space, many aspects of a problem such as work processes and goals are investigated and analysed. In the software world, high level requirements are extracted from the refined work processes which are then cultivated and organized into functional requirements for the desired software. Therefore, this research can be seen as problem focused and will help relate the real world with the software world.

6.2.4 A light weight method

The method is meant to teach students the process of problem structuring and analysis in RE course. The method needs to be light weight due to several reasons. First of all, there is a limited amount of time to complete the RE course taught in universities i.e. around three to six months. In addition, this method only covers one aspect of requirements engineering, so only a portion of the overall time would be available to teach this method. Secondly, it is assumed that being undergraduates, the students do not have any industrial experience or background knowledge of RE. Therefore, they are unaware of the RE concepts used. Thus, this method is explicitly developed to be light weight in nature so that it would be easier for students to understand and apply it within the limited time and resources available.

This method is characterized as light weight because it follows one simple structure and easy to apply procedure, and uses terms that students are already familiar with. In order to facilitate the understanding and application of the L-Soft method, the students are provided with a step-by-step guide on performing the problem structuring and analysis process, a glossary of terms used in the method, a sample of solved case studies and an automated tool to support the learning process.

6.2.5 The flow of L-Soft constructs

The constructs are the underlying concepts on which the working of L-Soft method is based on. The working of the L-Soft method is based on several constructs which are problems, causes of problems, work processes, goals, features and requirements. In the first step of L-Soft method, first four constructs that are problems that need to be solved, possible causes of problems, work process with which causes are connected and desired goal of each work process are identified. In the second step, each work process is analysed and presented as a set of steps in order to show its working. In the third step, the existing work processes are organized and change in work process is proposed based on its desired goal. In the fourth step, features that the product should support are extracted from refined work processes and divided into functional and non-functional requirements.

Even though the method shows one simple structure, it still allows for any amount of detail. The flow of the construct derivation is shown in Figure 6.1.

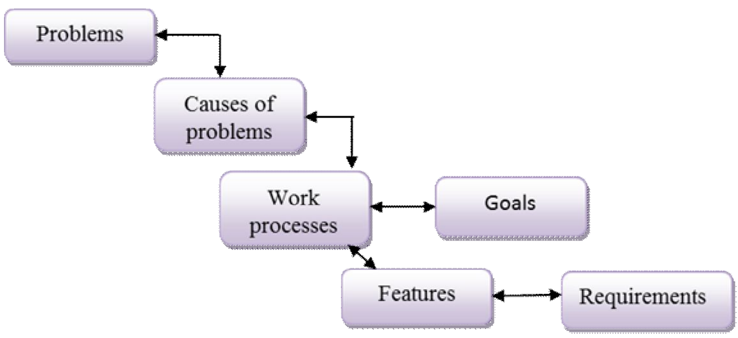


Figure 6.1: The flow of the construct derivation

6.2.6 A work process oriented approach

The L-Soft method follows a work process oriented approach. The work process oriented approach tends to understand the problem by recognising and analysing the current work processes and address the problem by refining the work processes. The idea is to change the flow of the processes so that the problem can be addressed. Initially, the causes of problems are identified by connecting them to the work processes. Each work process is analysed and understood in detail to identify the changes required in order to address the causes of the problems. The requirements are also extracted from the refined work processes. So the work processes are used as a means to identify the causes of the problems, the desired changes and finally, the requirements. Therefore, the requirements resulting from the application of L-Soft are seen to be dependent on the work processes.

6.3 The L-Soft method

During the course of producing the method, earlier versions of the method were presented to RE experts including RE researchers and academicians and was refined continuously based on their reviews. First version of the method was sent to the RE experts and the refinements were done based on their comments. The second version of the method was then presented at the seminar attended by many RE experts, their comments (feedbacks and suggestions) were incorporated in the method. Finally the third version of the method is presented here.

The method consists of two main phases which are problem space and software world, and five steps, as shown in Figure 6.2. This section discusses the purpose of each step of the method, presents the guidelines provided to guide students on the procedure of performing the steps, and the template (if applicable) that needs to be filled as an output of the steps.

It should be noted that the process of problem structuring and analysis cannot be performed without involving customers and users, and eliciting their requirements using different elicitation techniques. However, because this method is intended to be used in teaching, requirements elicitation is out of the scope of this work. Therefore, the customer's role will be played by the lecturers throughout the course of method application. However, if the method is to be used on a real project, then the analyst has to communicate with the users and use different elicitation techniques throughout the method application.

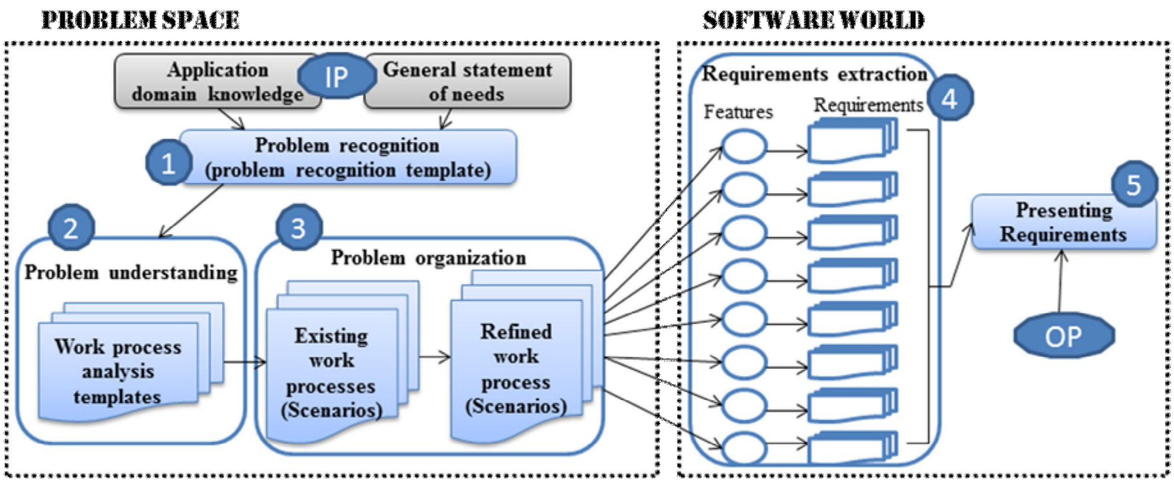


Figure 6.2: The L-Soft method

6.3.1 Input to the method

In order to apply the method, one must have a general statement of needs that includes customers' initial requirements and the domain knowledge which is the background information of the general area of interest. The general statement of needs is a short, quantified statement that states what the product is intended to do and the advantages it brings to the business, explains why the business is investing in the project and the benefits it wants to achieve, and justifies the project (Robertson & Robertson, 2012). This information serves as an input to the method.

6.3.2 The Problem space

(Sidky et al., 2002) suggested that problem analysis should include two main activities which are problem identification and problem decomposition. However, because the problems are located in the real world, in order to identify the problems, we should understand the world and its phenomena. Therefore, the problem space phase of L-Soft includes the steps that identifies, analyses and decomposes the problem in terms of the work processes that are carried out in the real world. This phase includes three steps: problem recognition, problem identification and problem organization.

6.3.2.1 Problem recognition

The problem recognition process identifies and describes the needs of a system for certain purposes (Yeh & Zave, 1980). From the general statement of needs and domain knowledge, it verifies the subject of analysis and identifies those problems that cause the project to start.

In order to understand the problems, the root cause(s) of the problems are identified. That root cause actually represents the problem (Sidky et al., 2002). The problem may have several causes. Problems that occur in the organization are connected to the work processes that are carried out there (Andersen & Fagerhaug, 2006). Therefore, the work processes that cause the problems, and the stakeholders who are responsible for performing those work processes are identified. The functional goals of each work process (that is the desired result intended from the change in each work process) are set with the expectation that fulfilling these goals can address the problems. These goals are the highest level requirements (Robertson & Robertson, 2012).

Finally the benefits (business goals) desired from the new/updated system are described. These are the business reason for the system such as to help the organization stay competitive in the marketplace, to provide better customer service, to automate certain functions, to comply with government legislation, or to meet any other variety of environmental demands. (Kulak & Guiney, 2004).

The guidelines for performing this process and the template to be filled as an output of this step are presented in Figure 6.3.

Problem recognition Guidelines

1.1 Perform an initial analysis on the subject and general statement of needs and extract the following information:

a) The customers' main **problems** that need to be solved by the development of new/updated software product. These are the reasons of the doing the project.

b) The possible **causes** behind the existence of the problems.

c) The work processes connected to causes that needed to change in order to address them.

d) The **stakeholders** involved in performing the work processes.

e) The **functional goals** of the stakeholders that should drive the changes in the work processes. Specify goals for every work process.

f) Write down the key benefits desired from new/updated system. These represent the **nonfunctional goals** of the new/updated software product.

1.2 Fill problem recognition template with the extracted information.

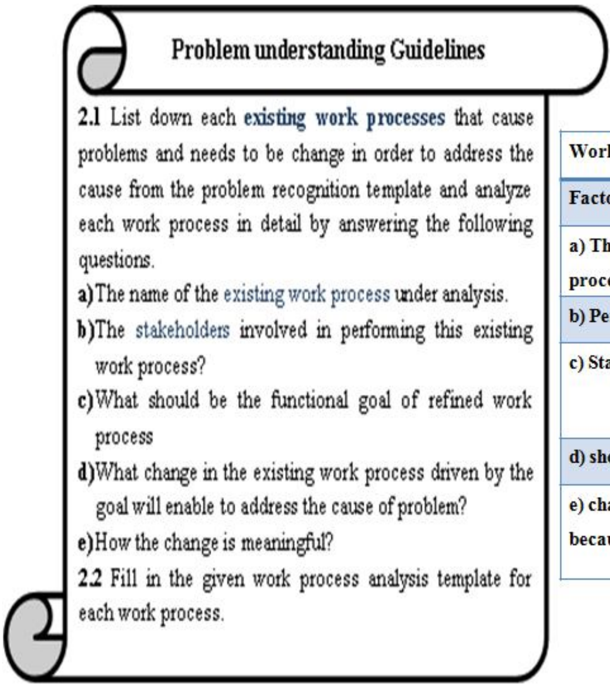
Problem recognition template	
Factor	Description
Subject of analysis	*Subject on which the analysis is performed*
a) The problems	*Describe the main problems*
b) Causes of problem	*Describe the causes of this problem*
c) Work processes	*Describe the work processes to which the causes are connected.*
d) stakeholders	*List down the stakeholders involved in performing the work processes.*
e) the functional goals of stakeholders	*The primary goals of the stakeholders (functional goals)*
f) Benefits of new/updated system	*List of key benefits (non-functional goals)*

Figure 6.3: The guidelines and template for the problem recognition step

6.3.2.2 Problem understanding

Problem understanding involves getting to the bottom of the problem. Because this method tends to understand the problem in terms of the work processes carried out in the real world and addresses the problem in terms of the changes that are required in the work processes, each work process in analysed in detail by identifying the stakeholders involved, the functional goal of the work process that should drive the change in the work process and how that change is meaningful. The work process that describes the current workings is termed as "existing work process".

The guidelines for performing this process and the template to be filled as a result of this process are presented in Figure 6.4. The template should be filled for each work process.



Work process analysis template	
Factor	Description
a) The existing work process	*The work process connected to the cause*
b) Performed by	*The stakeholders who perform work process*
c) Stakeholders' goal	*The functional goal of stakeholder that should drive the change in the existing work process*
d) should be changed to	*The refined work process driven by the goal*
e) change is meaningful because	*The reason why this change is meaningful to address the problem*

Figure 6.4: The guidelines and template for the problem understanding step

6.3.2.3 Problem organization

The problem can be fully understood only when it is decomposed and organized logically. A good structure helps organize the problems, making it easier to see what is necessary, where everything belongs, how to extract the requirements, and how the different requirements fit together (Alexander & Stevens, 2002).

In this step, the problems are structured as a family of scenarios in the form of work processes. In order to understand the work currently being done and the work that stakeholders desire to do, the analyst has to communicate with the customers in order to inquire about their work by using different elicitation techniques, or through observing their work by sitting with the users at the users' workplace.

Existing work processes are decomposed into a set of steps and are represented using textual scenarios, with the steps that cause problems highlighted. In order to address the problems, the highlighted steps are changed into an alternative way of performing the work processes; the change should be driven by the functional goals of the work processes. The work processes that depict the desired changes in order to achieve the functional goals are termed as “refined work processes”. The refined work processes are represented using graphical scenarios. The graphical scenarios facilitate the extraction of requirements as they can clearly show the features that the new/updated system should perform.

The guidelines for performing this process are presented in Figure 6.5.

Problem organization Guidelines

3.1 Represent each **work process** using work **scenario**/story that describes how the process is being performed currently. In order to get the details of work process from the users, different elicitation activities can be used.

3.2 In each **existing work scenario**, identify and highlight the activities that cause problem to occur in the process and need to be changed in order to address the problem.

3.4 Analyze highlighted activities and find out the alternate way to perform the activities driven by the goals of the project and record the alternate way in the refined work process. The **refined work process** should not change the nature of work but only the way work is done.

3.5 Write down **refined work process** for every existing work process. Use graphical scenarios (drawn using activity diagrams) to represent the refined work processes that can address the problems. The graphical scenario should include all the activities of work process (including those that need to be changed as well as those that shall remain same).

Figure 6.5: The guidelines for the problem organization step

6.3.3 The software world

This phase involves the extraction, analysis and organization of the requirements for the desired software. It is more towards requirements generation and analysis. The software world describes what the required software product shall do and define system that can be deployed to address the problem (Leffingwell & Widrig, 1999). It consists of two steps: requirements extraction and presenting requirements.

6.3.3.1 Requirements extraction

From the refined work processes, the features for the desired software product are identified and each feature is then translated into one or more software requirements. Features are the requirements at a higher level that the product shall support. Feature level requirements should not offer details as to what functions are needed in order for the product to support a feature; rather the requirements should be an abstract description of the feature itself (Gorschek & Wohlin, 2006). Each feature is decomposed into one or more requirements that represent one user task (at the lowest level).

The function level is a repository of functional requirements, i.e. what a user should be able to do (actions that are possible to perform) and also for non-functional requirements. Function level requirements are detailed and complete enough to be handed over to a system designer for further evolution and finally be a basis for the design (Gorschek & Wohlin, 2006). Components of an imaginary requirement in traditional style are user type, result type (verb), object and qualifier (adverbial phrase (optional)) (Alexander & Stevens, 2002). Typically, requirements are specified in lists and expressed in terms of "the system shall" (Kulak & Guiney, 2004). All requirements are broken down until functional level where they are good enough for initiating a development effort (project) and can act as a basis for design (Gorschek & Wohlin, 2006).

The guidelines for performing this process and the template to be filled with the extracted features and requirements are presented in Figure 6.6.

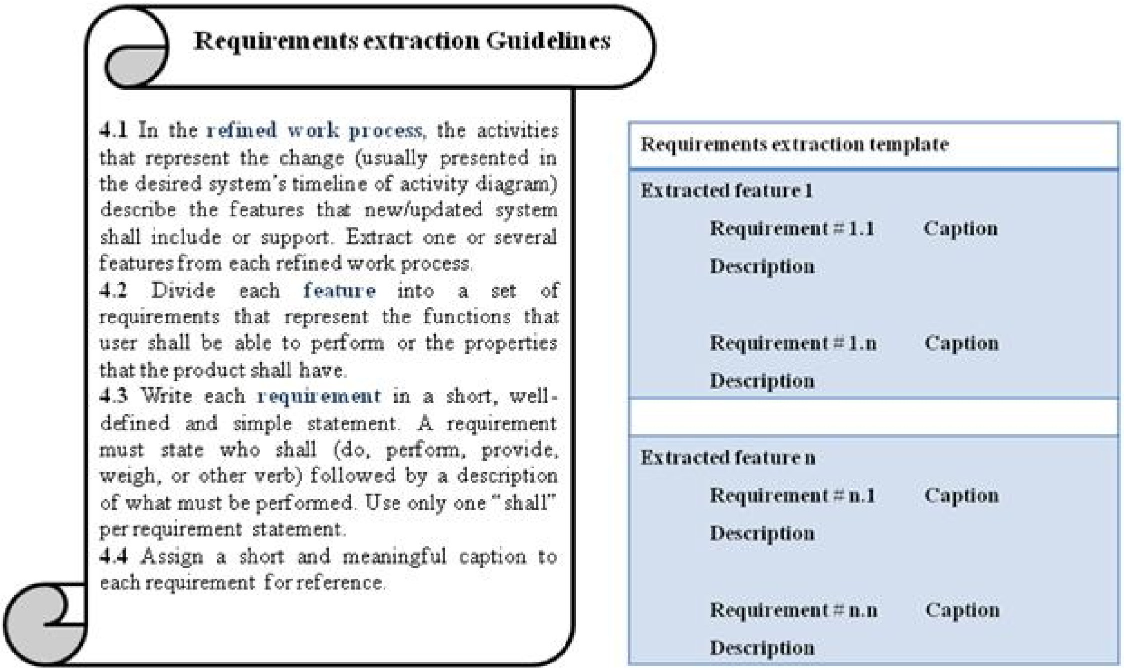


Figure 6.6: The guidelines and template for the requirements extraction step

6.3.3.2 Presenting requirements

A set of extracted requirements is the output from the previous step. However, this set of requirements must not be just a pile of requirements texts. Each requirement is unique and was written by someone at a particular time, and may have been modified similarly or reviewed, and should have a priority. A complete requirement consists of a text and all of this status information (Alexander & Stevens, 2002). Well-written requirements prevent common but serious problems. Therefore in this step, requirements with some of the status information are presented. This step involves assigning types and priorities to each requirement and checking them using an analysis checklist.

The requirements are either functional or non-functional types. Functional requirements are what the user needs for the system to work. These are requirements we typically think of when we describe systems. Non-functional requirements are global, fuzzy factors or principles that relate to the system's overall success. The best time to identify non-functional requirements is while you are exploring the functional requirements. They are about response time needs and what annoys users about the way the system works now. The answers will give valuable leads for non-functional requirements. These non-functional requirements contribute to the usefulness of a system (Kulak & Guiney, 2004).

A priority is assigned to each requirement to indicate how essential it is to include it in a particular product release. The term "priority" means, i) the state of being prior (i.e., given precedence in terms of date or time), ii) given or meriting attention before competing alternatives, and iii) given preference (Firesmith, 2004). A priority is assigned to each requirement based on its importance and implementation order. The requirements are either essential (high priority), useful (medium priority) or desirable (low priority) (Firesmith, 2004).

Finally, requirements are assessed using an analysis checklist which consists of a list of questions through which each requirement is assessed to ensure acceptable quality. Each requirement is checked individually. Each individual requirement must be clear, atomic, verifiable, and prioritized. But this, while helpful, is not sufficient. The requirements must also be checked as a set. The set as a whole must be realistic, consistent, and complete. Each requirement could seem sensible, but the total set could be impossible to implement (Alexander & Stevens, 2002). Each of these components for requirements contributes to the structure of the overall requirement, and all of them are important in order to understand the whole requirement (Robertson & Robertson, 2012).

The guidelines for performing this process and the analysis checklist are presented in Figure 6.7 while the template to be filled as the output of the complete problem structuring and analysis process is presented in Table 6.1. The template retrieves data from this step as well as from previous steps.

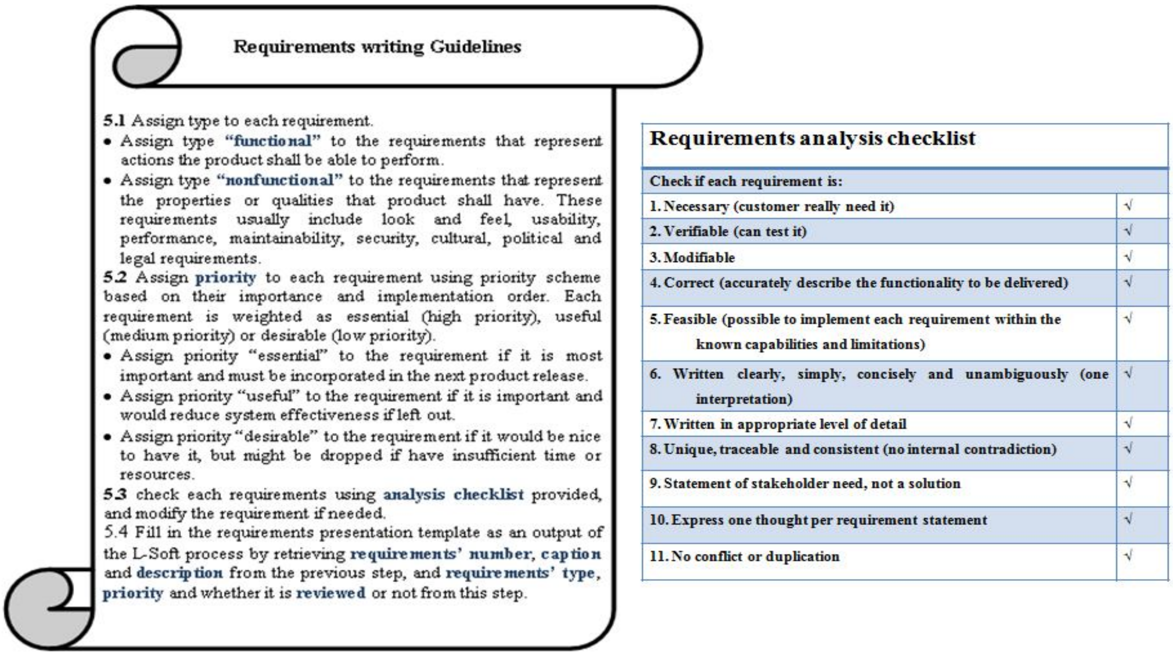


Figure 6.7: The guidelines for extracting requirements and the requirements analysis checklist

Table 6.1: The template for presenting the output of the L-Soft method

Requirements presentation template					
Req#	Caption	Description	Type	Priority	Reviewed
⋮	⋮	⋮	⋮	⋮	⋮

6.4 Summary

This chapter presented the L-Soft method in detail. Each step of the method has been discussed and the guidelines and templates have been presented as a supporting material that will enable students to learn and apply the method. The next chapter will present the L-Soft tool developed to support the application of the method and an exemplar (example problem) that will show its application.

Chapter 7- The L-Soft tool

7.1 Introduction

The focus of this chapter is on the implementation of L-Soft method, introduced in Chapter 6, into a software tool to support its teaching and learning. The tool is a web based system that will help students in learning and applying the L-Soft method and lecturers in teaching the method. The motivation behind developing the L-Soft tool is discussed in the next section, while the tool design is presented in Section 7.3. Section 7.4 deals with the implementation of the L-Soft tool.

7.2 Motivation for the L-Soft tool

As the L-Soft method was developed to facilitate students in learning and lecturers in teaching the process of problem structuring and analysis in the RE course, it was felt necessary to develop a prototype based on the L-Soft method to support the learning and teaching of the method. If no tool involved, L-Soft can still work and can be performed manually or using different tools but it may take more time and effort to apply the method.

Usually, tools appear as the means to automate a methodology (Avison, Golder, & Shah, 1992). They originally have very simple objectives of providing automated support to some previously manual tasks such as documentation, but it is becoming clear that they have a more fundamental effect. They are beginning to simplify methods, making various activities more consistent, and in some instances, shortening the process itself (Macdonald, 1987).

The L-Soft tool is aimed at providing two main objectives; first to automate the steps of the method and second, to provide learning support in order to facilitate users to learn, understand and apply the method, and to store and retrieve a project's data.

7.3 The design of the L-Soft tool

The functional design of the L-Soft tool is presented in Figure 7.1.

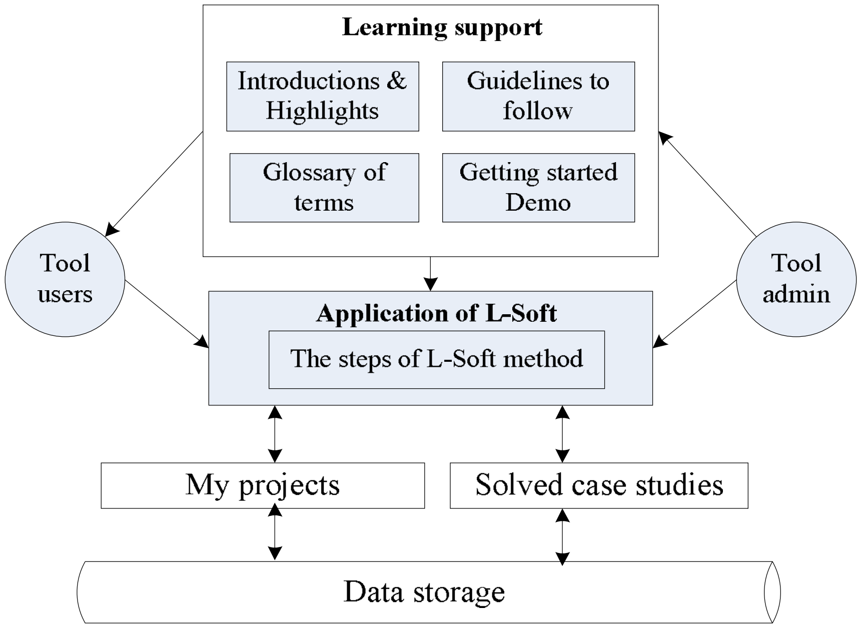


Figure 7.1: The functional design of the L-Soft tool

The functional design in Software engineering is a structured representation of the functions (activities, actions, processes, operations) within the modelled software (Jacobson, 1992). The functional design of L-Soft tool presents the functionality provided by the L-Soft method. The components in the functional design of the tool illustrated in Figure 7.1 are described in the following sections.

7.3.1 Users

The tool is meant to be used by students (tool users) and lecturers (tool admin). The tool users have to register first by providing a username and password. Once registered, they can login into the system and create new projects, or work on existing projects, or view the solved case studies uploaded by their instructor (tool admin) to provide learning support.

The tool admin, on the other hand, has administrative rights and can login into the system by using an admin username and password. Once logged in, the admin can check the projects done by the tool users, or create new projects and make them public (solved case studies for students) to support students' learning. The use case diagram showing the functions that an admin and user can perform in presented in Figure 7.2.

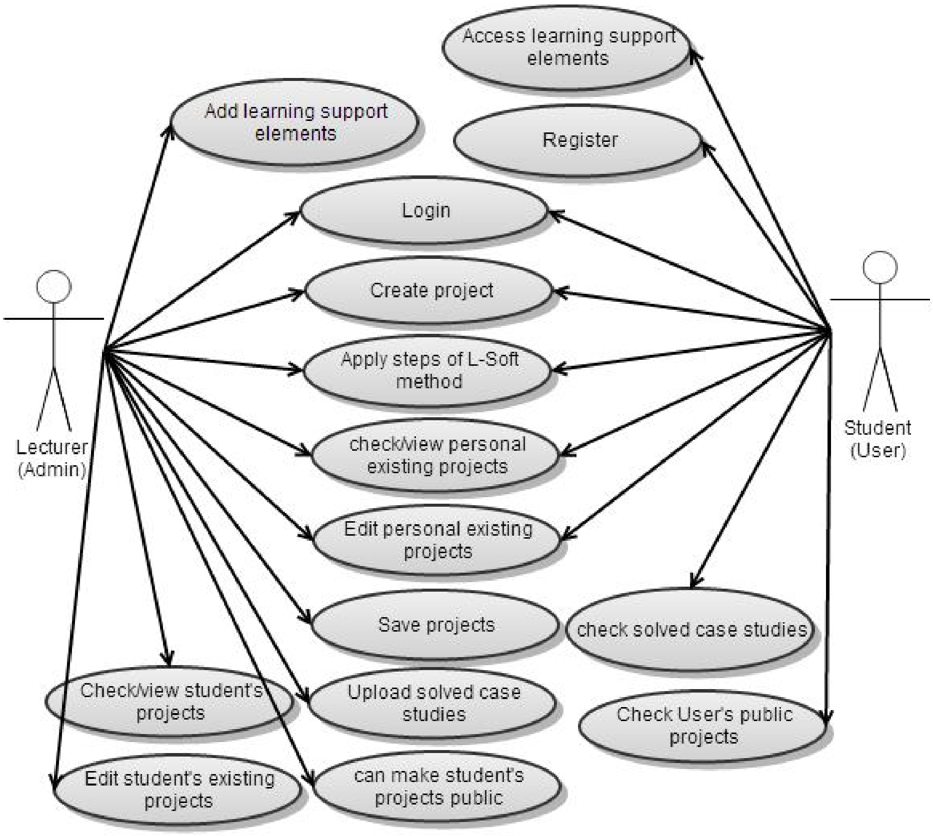


Figure 7.2: The roles of the tool admin and user

7.3.2 Learning support

As the tool was developed to support students in learning the L-Soft method, the tool users are provided with learning support in many forms such as: -

- the introduction and method's highlights which are provided so that the users can understand the L-Soft method's underlying concepts (see Figure 7.3)
- the glossary of terms which are provided so that the users can learn the definitions of the different terms used in the method (see Figure 7.4)

- the “getting started” demo which has been provided to allow the users to get familiar with the working of the tool (see Figure 7.3)
- Guidelines which have been provided with each step of the method to facilitate the users with the proper procedure of applying the L-soft method (such as presented in Figure 7.6 and 7.7).

Figure 7.3 shows the screen shot of the homepage of the L-Soft tool. The introduction, highlights, demo and steps of the L-Soft method are shown by the legends. Figure 7.4 shows the ‘”Glossary of the terms” page.

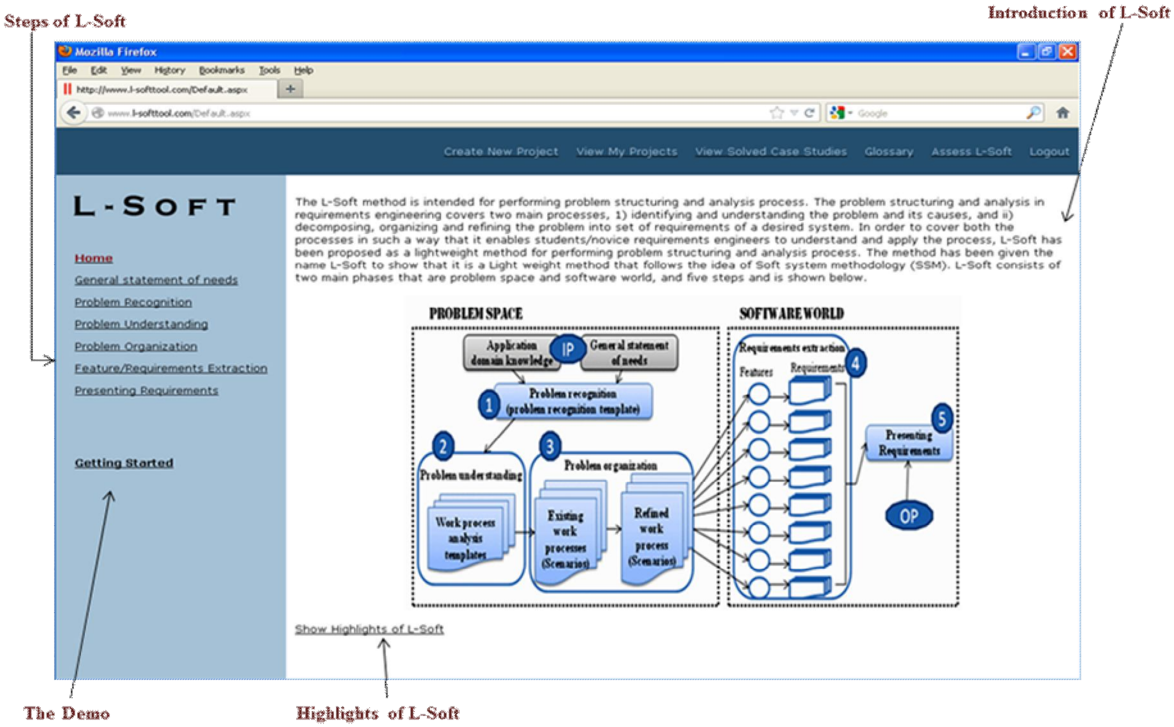


Figure 7.3: The screen shot of the homepage of the L-Soft tool

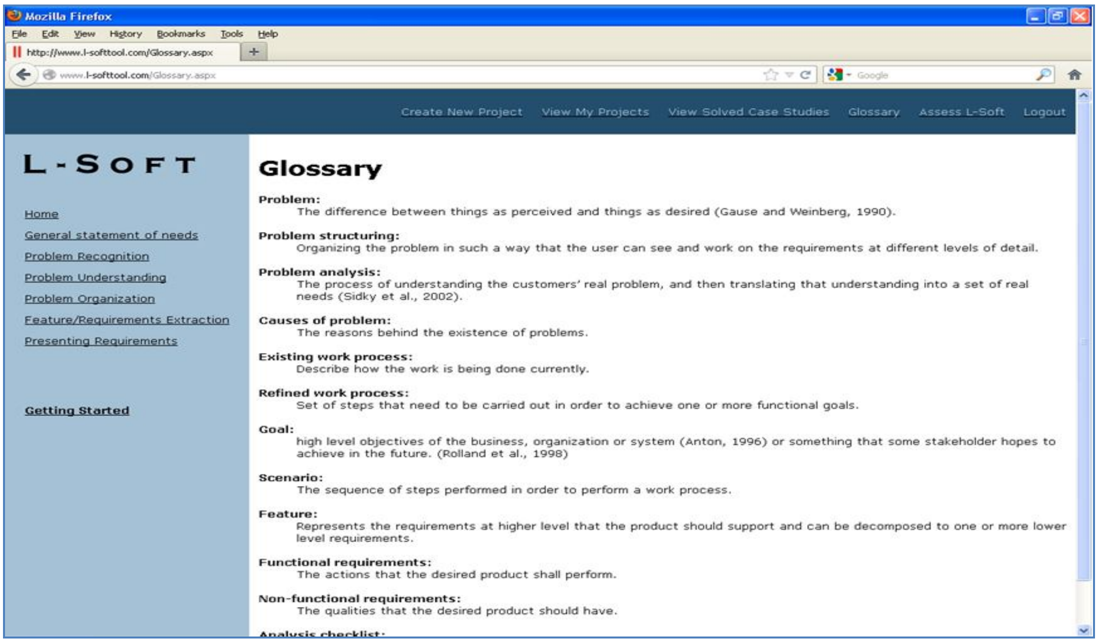


Figure 7.4: The screen shot of the "Glossary of terms" page of the L-Soft tool

7.3.3 Application of L-Soft

The tool facilitates the application of the steps of the L-Soft method. The steps are shown in the left pane of the homepage (see Figure 7.3). The users can work on and save many projects, and can read the guidelines provided with each step and apply the steps of the method accordingly. The data entered in one step will be shown to users in other steps (if needed) to facilitate the method application process. To start working on a project, the user has to create a new project, and add the project's name and general statement of needs as an input to the project before saving it.

In this section, the screen shots of all the steps are presented. A solved case study entitled "Employee training system (ETS)" is presented as an example to show the method application. Figure 7.5 shows the screen shot of the "General statements of needs" page (input to the method) of the ETS project. This page appears when the user either clicks the "Create new project" button or loads a previously saved project. For the case of the former, this page allows the user to add a project name and a general statement of needs for the project before saving it.

As for the latter, it allows the user to alter the data or just view the project's general statement of needs. A user can also go back to this page while working on the project by clicking the "General statement of needs" link.

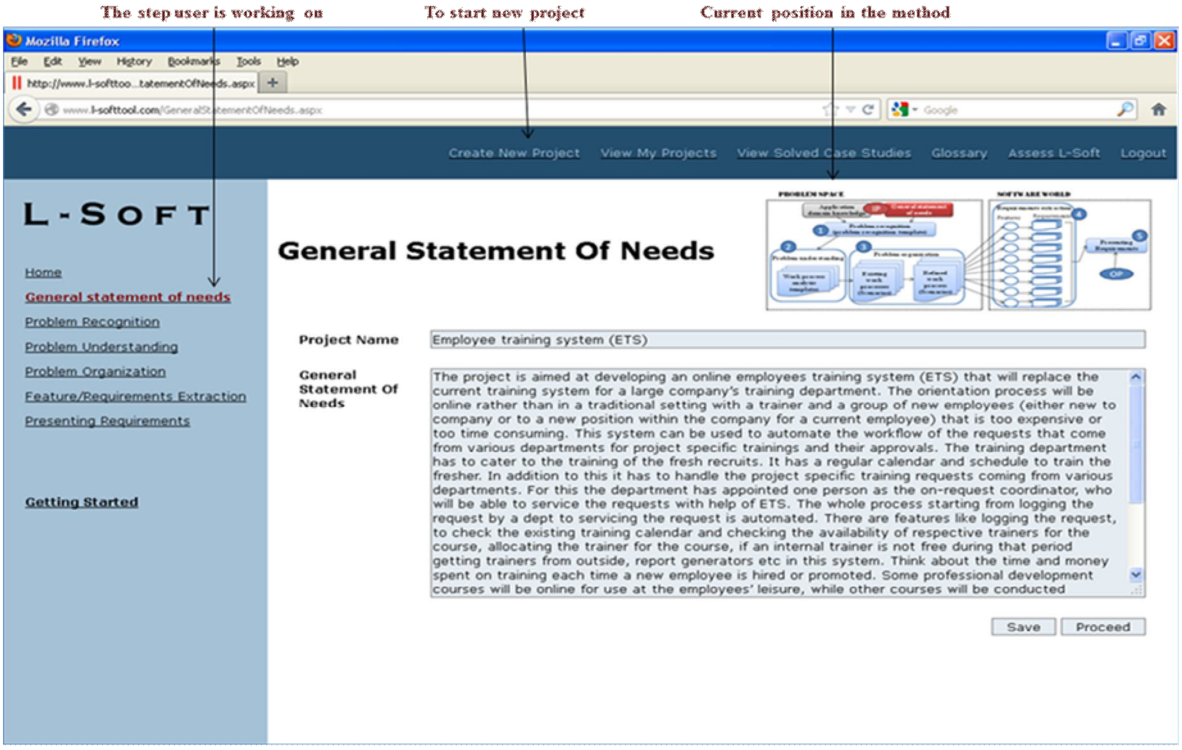


Figure 7.5: The screen shot of the "General statement of needs" (method input) page

In every step, the graphical representation of the L-Soft method is presented which shows the position of the user within the method by highlighting the current step in red. The current step in the left pane is also highlighted in red.

The screen shot of the first step of the method i.e. "Problem recognition" for the ETS project is shown in Figure 7.6.

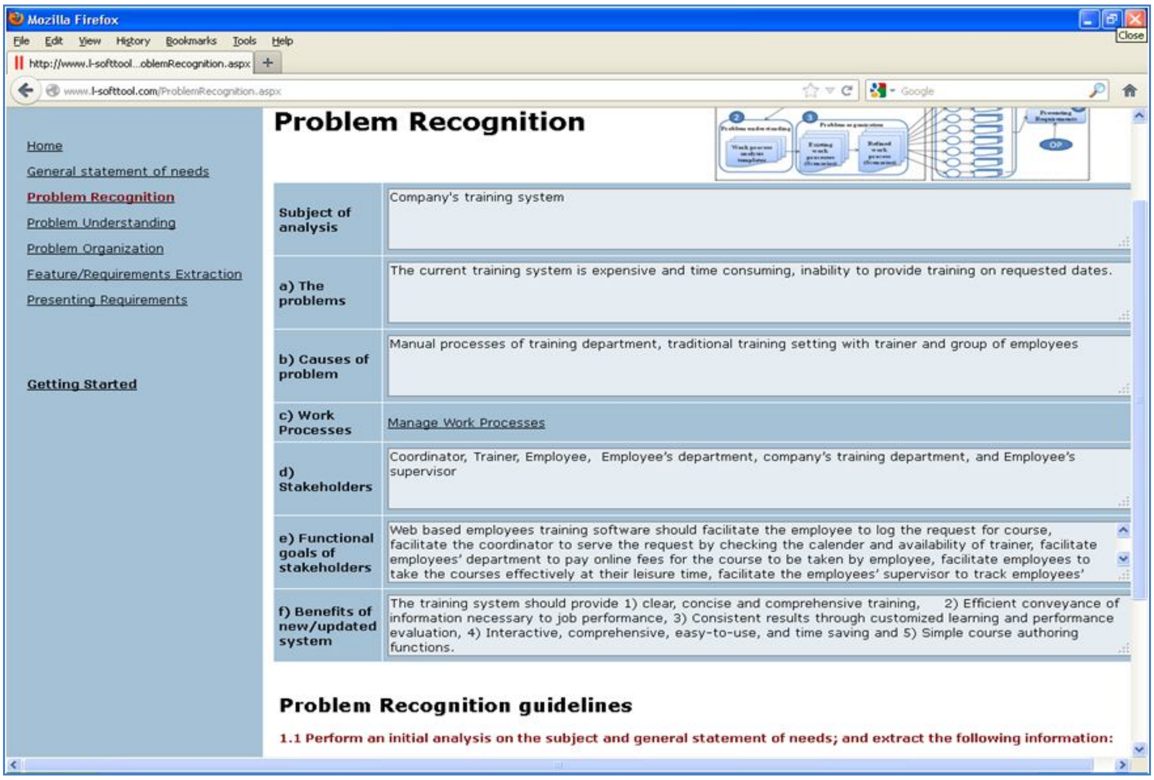


Figure 7.6: The screen shot of the “Problem recognition” page

In the problem recognition step, the work processes can be added, edited or deleted by clicking the “Manage work process” link. Figure 7.7 shows the screen shot of the work processes for the ETS project.

Figure 7.7: The screen shot of the work processes

The work processes entered in the problem recognition step will be shown in the "Problem understanding" page where the user has to analyse the work processes by clicking the "Analyse work process" link for each of the process. This will lead the user to the work process analysis template. The screen shot of the "Problem understanding" page is shown in Figure 7.8.

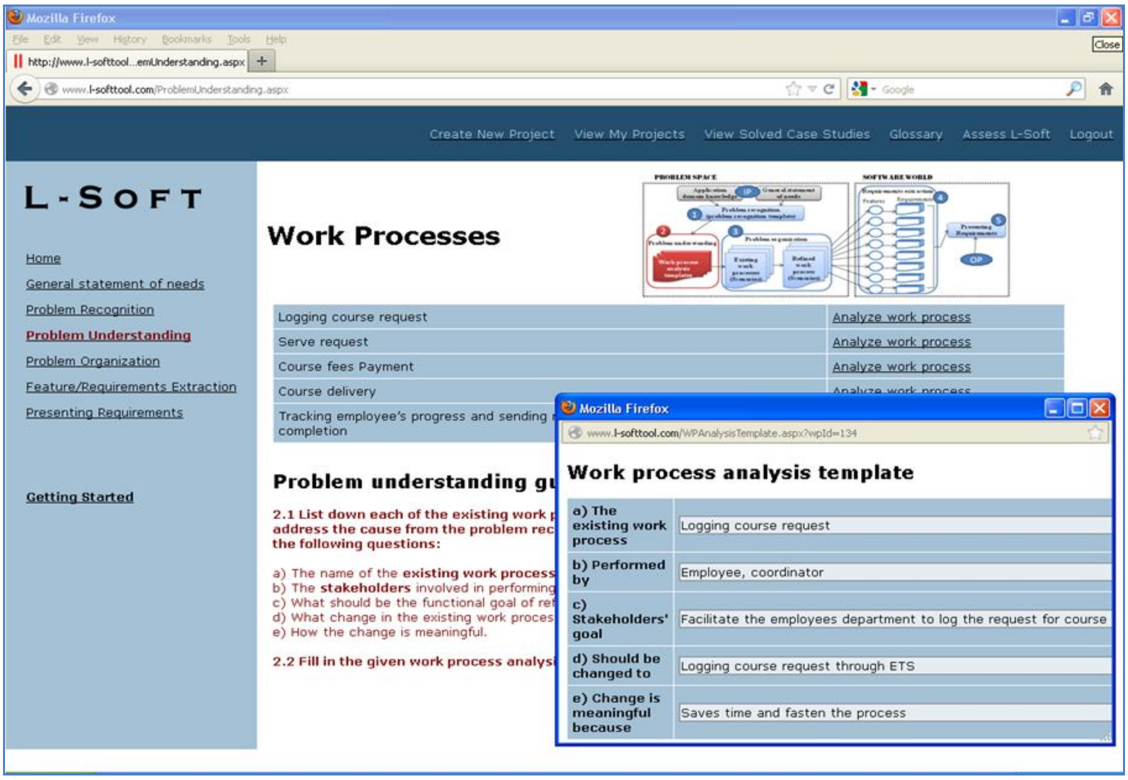


Figure 7.8: The screen shot of the "Problem understanding" page

In the next step i.e. problem organization, the work processes are presented and the user has to add the textual scenarios for each of the existing work processes and upload the graphical scenarios for each of the refined work processes by clicking the "Add scenario" link. The screen shot of the "Problem organization" page is shown in Figure 7.9 and the "Add scenario for existing work process" and the "Add scenario for refined work process" pages are shown in Figure 7.10.

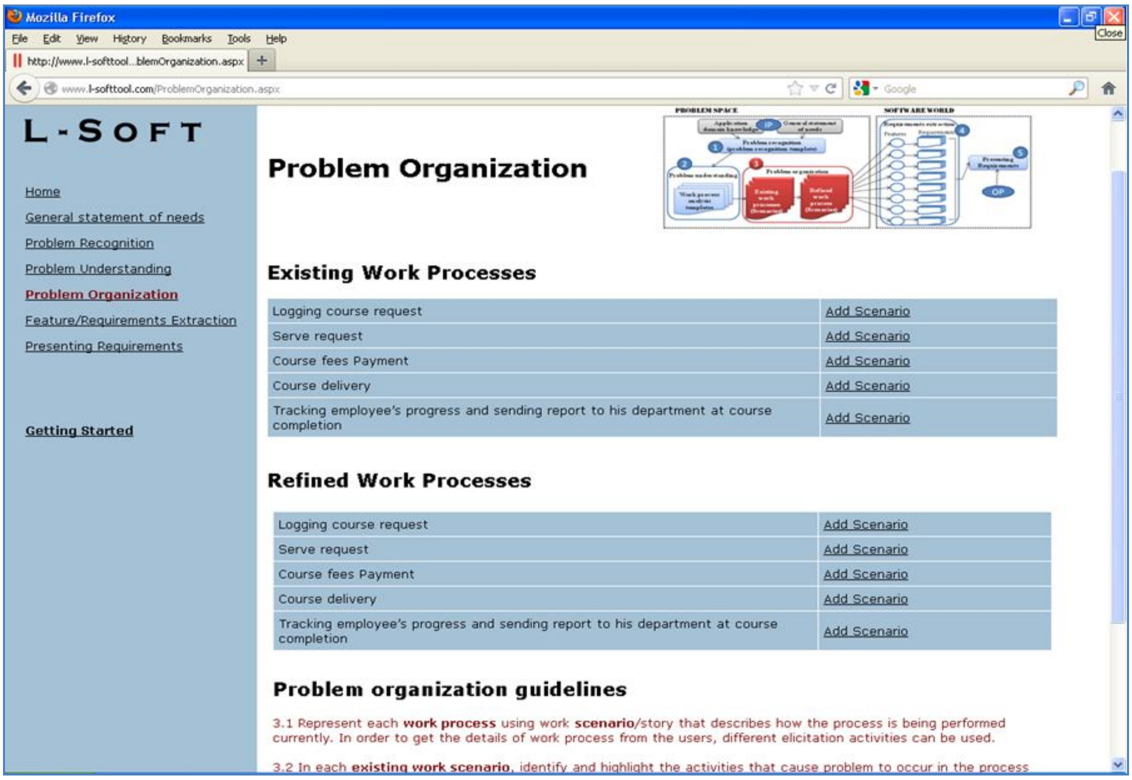


Figure 7.9: The screen shot of the "Problem organization" page

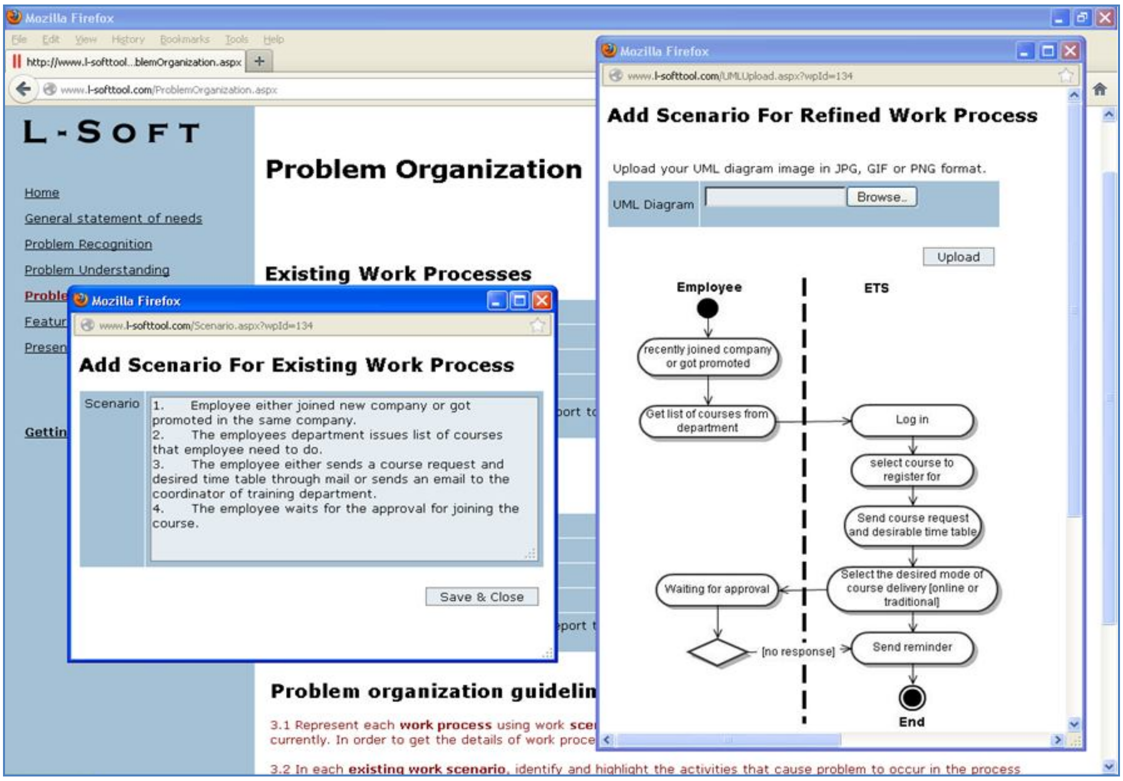


Figure 7.10: The screen shots of pages for adding scenarios

In the next step i.e. feature/requirements extraction, the user can add/edit/delete the features and add/edit/delete the requirements. The screen shot of the "Feature/requirements extraction" page is shown in the Figure 7.11.

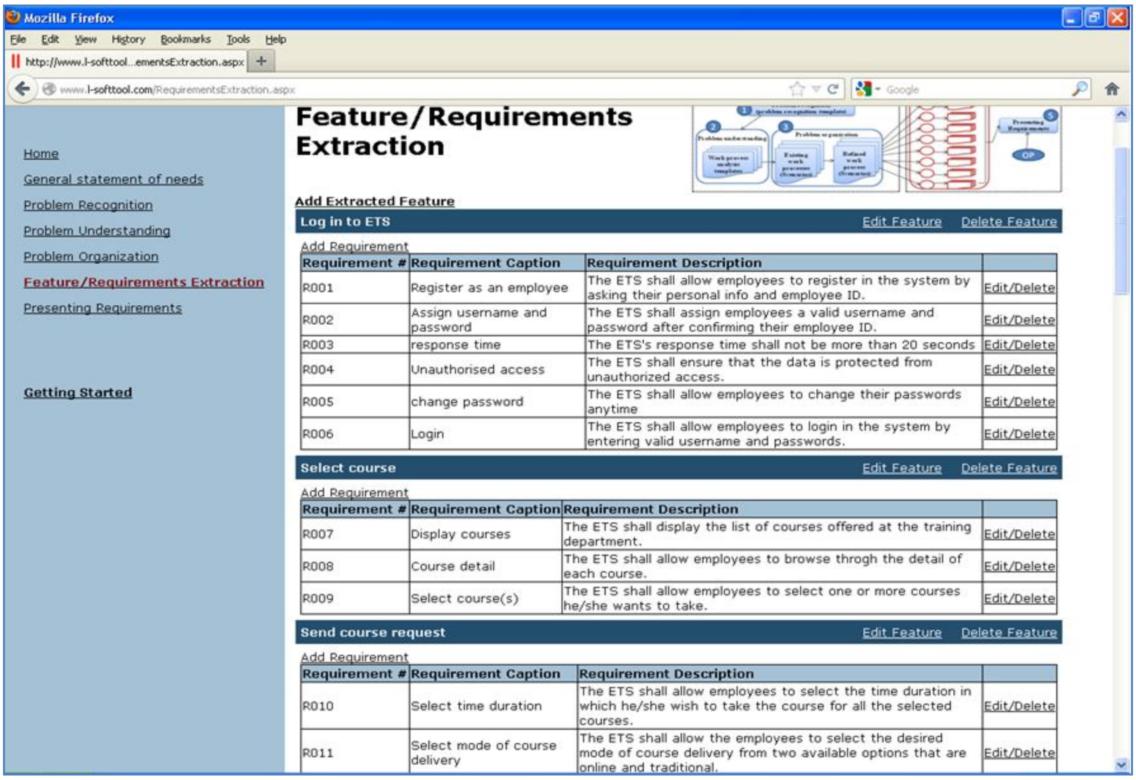


Figure 7.11: The screen shot of the "Feature/requirements extraction" page

In the final step, the requirements added in the feature/requirements extraction are presented to the user. The user then has to assign a type and priority to each of the requirements before reviewing them. By clicking the "Review" link, the requirement analysis checklist appears which the user has to check in order to review the requirements. The screen shot of the "Requirements presentation" page is shown in Figure 7.12 and the requirement analysis checklist is shown in Figure 7.13.

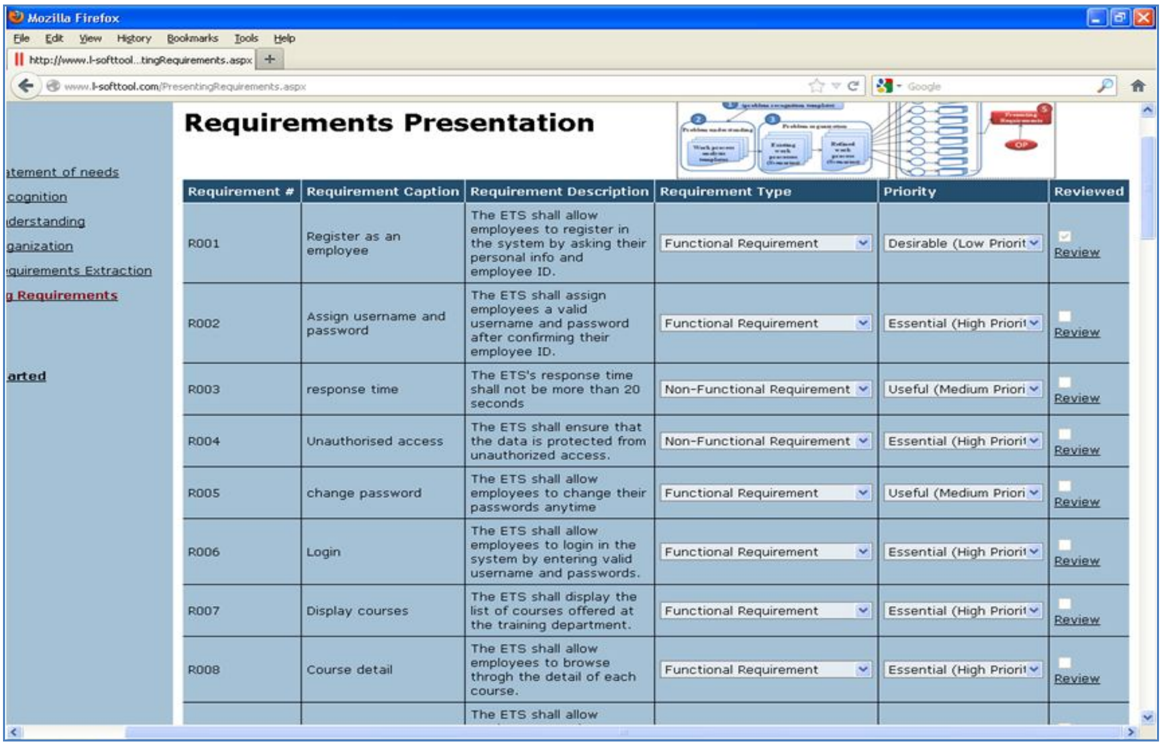


Figure 7.12: The screen shot of the “Requirements presentation” page

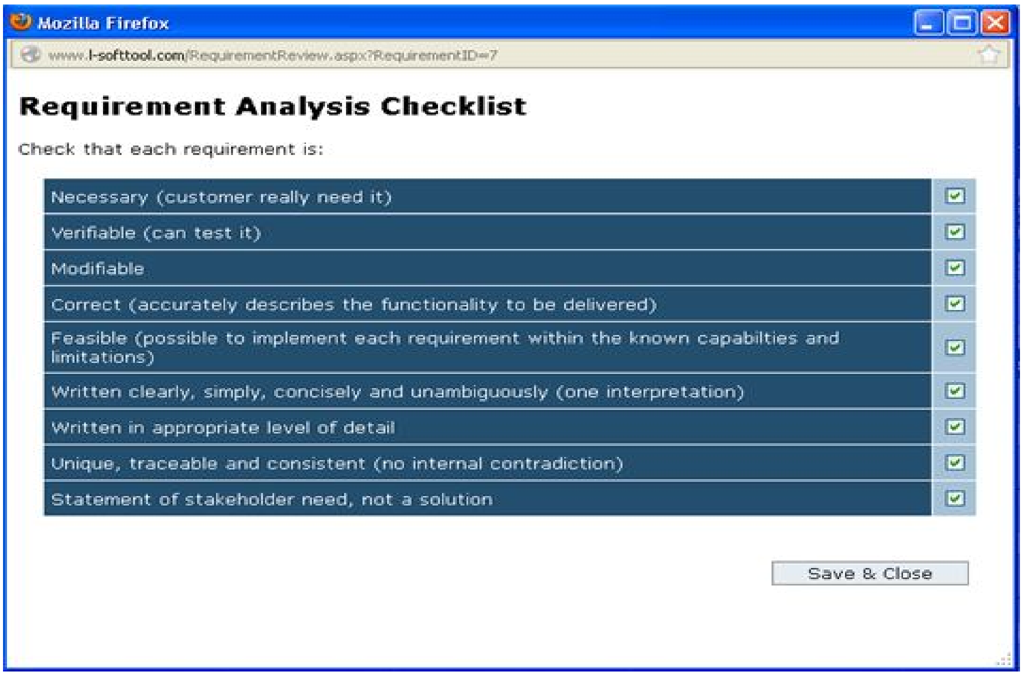


Figure 7.13: The screen shot of the requirements analysis checklist

7.3.4 Solved case studies

The solved case studies consist of the example problems solved by the admin by applying the steps of the L-Soft method to facilitate in the method application. The user can load and view the complete solved case studies and follow them, while the admin can add, edit or delete case studies. The screen shot of the "Case studies" page that appears by clicking the "View solved case studies" link is shown in Figure 7.14

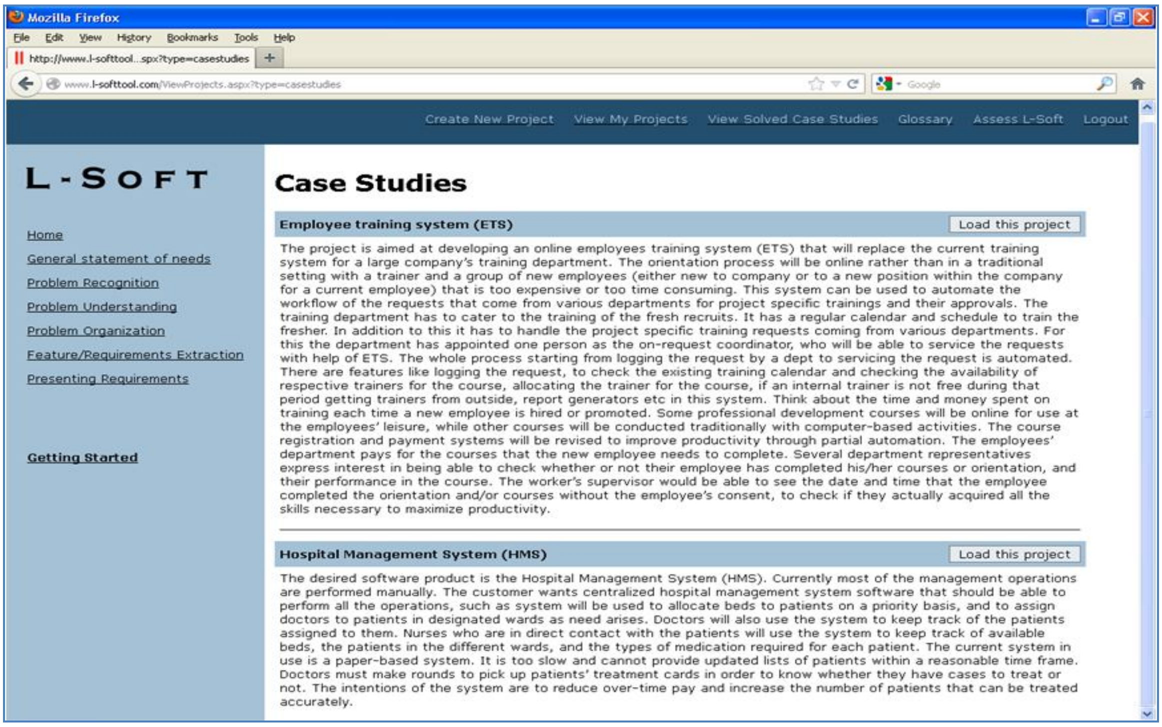


Figure 7.14: The screen shot of the "Case studies" page

7.3.5 My projects

The projects created by the user are saved in "My projects". The users can add and edit many projects, or work on a saved project. To work on an already saved project, the user has to first load the project. The screen shot of the "My projects" page is shown in Figure 7.15.

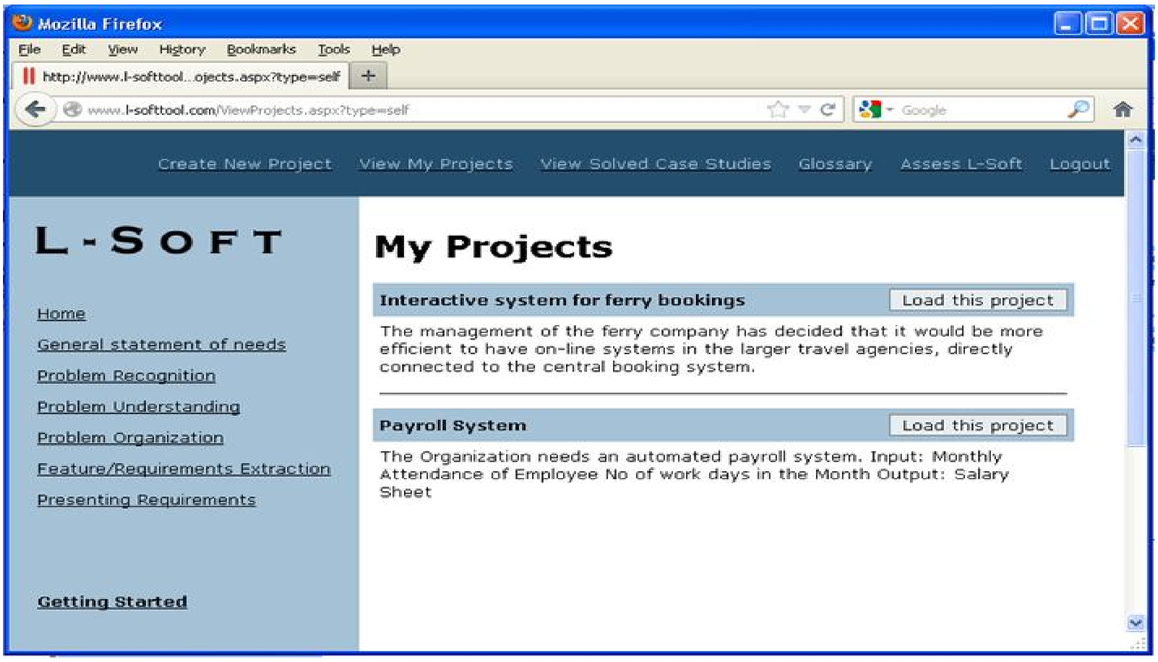


Figure 7.15: The screen shot of the "My projects" page

7.4 Implementation

The L-Soft tool has been implemented using the C# language version 4.0 and the Microsoft SQL server database on the ASP.net Platform. Dot NET is a powerful platform which provides a rich set of classes, while C#.NET is capable of powerful integration with Microsoft Office, Outlook, and SharePoint etc. In addition, Microsoft SQL server was chosen because of its scalability (in terms of records) and more powerful administration and integration with other Microsoft products. In the future, should this application be extended to an enterprise application, both C#.Net and Microsoft SQL server will still be chosen as the programming language of choice.

The L-Soft tool is built as a web-based database application because:

1) We wanted it to be accessible from anywhere by any number of users.
2) Each user can have his/her own account, in which he/she would be able to work on his/her own case studies.
3) Case Studies can be saved into the database, which will then be accessible by the same user or other users.

4) The application is to be centrally hosted in one place, which would be accessible by users across the network (and the biggest network accessible by the browser is the Internet), therefore it was built as a browser based application.

7.5 Summary

This chapter presented the functional design of the L-Soft tool built based on the L-soft method. Each part of its functionality is explained in detail and the screen shots of the web based L-Soft tool are presented. The L-Soft tool automates the steps of the method as well as provides learning support to students. The implementation details of the tool are also explained.

Chapter 8- L-Soft validation

8.1 Introduction

In this chapter, the L-Soft method was evaluated using three different validation methods, which are feature analysis, method acceptance testing and formal experiment. Feature analysis evaluates the appropriateness of the L-Soft method for performing the problem structuring and analysis process. Method acceptance testing, on the other hand, evaluates the likelihood of adopting the L-Soft method for teaching students problem structuring and analysis as part of the RE course in universities. Finally, formal experiment evaluates the ability of L-Soft to teach students the process of problem structuring and analysis. The overall validation framework and the details of each of the validation technique are presented in the following sections.

8.2 Selection of the validation methods

The validation methods were selected using a methodology named DESMET (Kitchenham, 1996), which is a methodology for evaluating software engineering methods or tools. DESMET identifies nine validation methods and a set of criteria to help the evaluator choose the most appropriate one for his needs, based on the different evaluation contexts that the evaluating team may have to deal with. These nine validation methods are listed in the table 8.1.

Table 8.1: The validation methods for evaluating software engineering methods/tools (Kitchenham, 1996)

No.	Evaluation type	Description
01	Quantitative experiment	An investigation of the quantitative impact of methods/tools organised as a formal experiment
02	Quantitative case study	An investigation of the quantitative impact of methods/tools organised as a case study
03	Quantitative survey	An investigation of the quantitative impact of methods/tools organised as a survey

Table 8.1, continued

04	Qualitative screening	A feature-based evaluation done by a single individual who not only determines the features to be assessed and their rating scale but also does the assessment
05	Qualitative experiment	A feature-based evaluation done by a group of potential users who are expected to try out the methods/tools on typical tasks before making their evaluations
06	Qualitative case study	A feature-based evaluation performed by someone who had used the method/tool on a real project
07	Qualitative survey	A feature-based evaluation done by people who have had experience of using the method/tool, or have studied the method/tool
08	Qualitative effects analysis	A subjective assessment of the quantitative effect of the methods and tools, based on expert opinion
09	Benchmarking	A process of running a number of standard tests using alternative tools/methods (usually tools) and assessing the relative performance of the tools against those tests

According to DESMET, an object to be evaluated can be a generic method, a specific method or a tool (Kitchenham, 1996). In this research, the evaluation object is a specific method that is the L-Soft method. The objectives of the L-Soft validation were compared to the validation methods presented in DESMET, and those that are appropriate for our evaluation requirements were chosen and the evaluations were performed using those methods.

The objectives of the validation process and selected validation methods are presented below.

Objective 1: To evaluate whether the L-Soft method is appropriate to be used for performing the problem structuring and analysis process.

Validation method 1: Feature analysis (Qualitative survey in Table 8.1) was performed to evaluate the appropriateness (i.e. how well the method fits the needs) of the method.

The software engineers and RE lecturers studied/applied the L-Soft method and assessed the degree to which the identified features are supported by the method.

Objective 2: To evaluate whether the L-Soft method can be used to perform the problem structuring and analysis process and can be suitably adopted to teaching it to students as part of the RE course in universities, and to highlight the areas of deficiencies of L-Soft.

Validation method 2: Method acceptance testing (Quantitative survey in Table 8.1) was performed on software engineers and RE lecturers who have studied/applied the L-Soft method in order to gather their perception on the method.

Objective3: To evaluate whether the L-Soft method is a light-weight method that can enable students to perform the problem structuring and analysis process and can be adopted in teaching students as part of the RE course in universities.

Validation method 3: Formal experiment (Quantitative experiment in Table 8.1) was performed on undergraduate software engineering students in the classroom.

The second and third validations were performed based on an evaluation model called Method Evaluation Model (MEM) presented in the following sub-section.

8.2.1 Method Evaluation Model (MEM)

The Method Evaluation Model (MEM) was chosen as a basis to perform the validation method 2 (method acceptance testing) and 3 (formal experiment). MEM is a theoretical model proposed by Moody (D.L. Moody, 2003) for the validation of Information Systems (IS) design methods. According to Moody, methods or “knowledge how” define ways of doing things, therefore an entirely different approach is required to validate methodological knowledge because a method does not describe any external reality, so it cannot be true or false, only effective or ineffective. Thus, the validity of the method can only be established by its applicative success in practice (D.L. Moody, 2003).

Moody's model is based on the theory that actual efficiency and effectiveness that determine intentions to use a method are only 'second-hand', via its perceived ease of use and its perceived usefulness. This is due to the fact that in human behaviour, subjective reality is more important than objective reality (España, Condori-Fernandez, González, & Pastor, 2010). Along with the validation of IS method, MEM can also be used to evaluate methods in other fields such as in (España et al., 2010) MEM has been used for evaluating a requirements engineering method.

In this research, Moody's MEM (D.L. Moody, 2003) is refined according to the evaluation objectives (2 and 3) and presented in Figure 8.1.

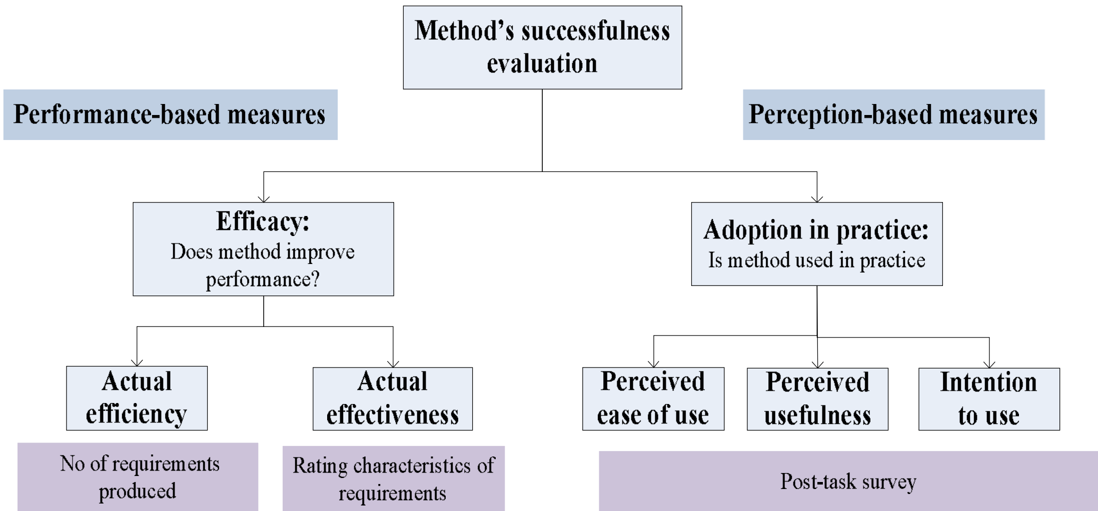


Figure 8.1: The refined Method Evaluation model

The presented MEM is called refined MEM due to several reasons. The Moody's MEM (D.L. Moody, 2003) presents six constructs, that are, actual efficiency, actual effectiveness, perceived ease of use, perceived usefulness, intention to use and actual usage. In our refined MEM, the construct actual usage (the extent to which a method is used in practice) was not evaluated, as it was not possible to measure actual usage in an experimental context.

In addition, Moody's MEM (D.L. Moody, 2003) hypothesize the casual relationships between the constructs of the model such as perceived ease of use can be determined by actual efficiency, perceived usefulness can be determined by actual effectiveness and so on. In the refined MEM, these relationships were not taken into account as it was realised that the objectives can be met using five constructs.

The validations were performed to evaluate whether the method is successful in achieving its objectives. According to Moody (D.L. Moody, 2003), two dimensions of "success" need to be considered in evaluating methods which are efficacy and their adoption in practice. The performance-based measures are used to find the efficacy of a method while the perception-based measures are used to measure its adoption in practice.

8.2.1.1 Performance-based measures

How well did the subjects understand the method (D.L. Moody, 2002) and perform the evaluation tasks (D. L. Moody, Sindre, Brasethvik, & Sølvberg, 2003)? Actual efficiency and actual effectiveness are two dimensions of performance (D.L. Moody, 2003), in which actual efficiency is the effort required to apply a method (the measure of the number of requirements produced as an output), while actual effectiveness is the degree to which the method achieves its objectives (measured by rating the characteristics of the produced requirements).

8.2.1.2 Perception-based measures

How effective did the subjects perceive the method to be? (D. L. Moody et al., 2003).

Perceptions can be obtained by using three measures which are perceived ease of use (the degree to which a person believes that using a particular method would be free of effort), perceived usefulness (the degree to which a person believes that a particular method will be effective in achieving its intended objectives) and intention to use (the extent to which a person intends to use a particular method) (D.L. Moody, 2003). These three are measured through a post-task survey.

8.3 The validation framework

The overall validation framework is presented in Figure 8.2. It presents the three validation methods used in the research and how they are performed.

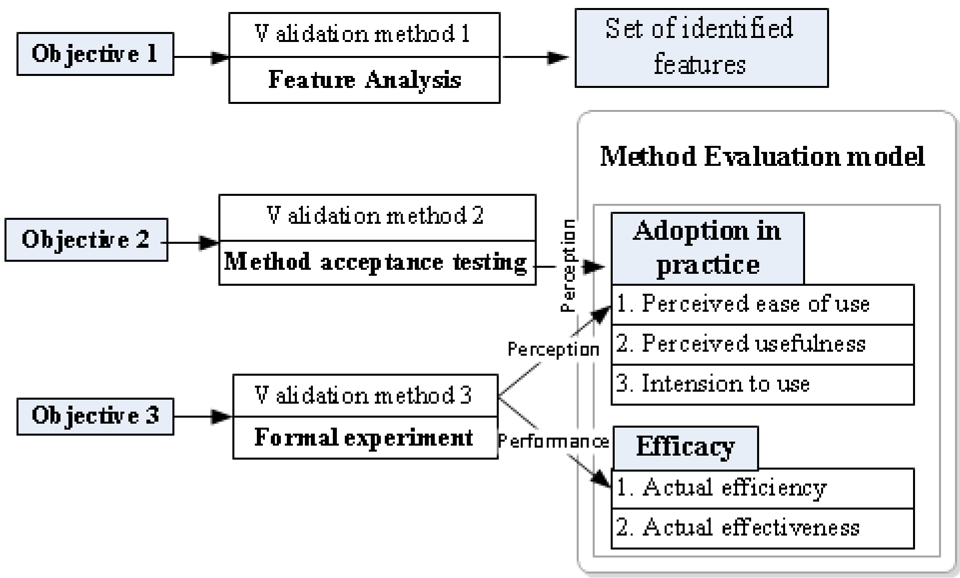


Figure 8.2: The validation framework

The Validation method 1, which is feature analysis, is performed in order to fulfil the first objective. The constructs for this validation are the set of identified features.

In order to fulfil the second objective, a method acceptance testing is performed using perception-based measures (i.e. perceived ease of use, perceived usefulness and intentions to use) of the MEM to measure the likelihood of the L-Soft method's adoption in practice.

Meanwhile, the third objective is fulfilled by performing a formal experiment using performance-based measures (i.e. actual efficiency and actual effectiveness) as well as perception-based measures of the MEM to measure the efficacy and likelihood of adopting the L-Soft method into practice. All three validations and their results are explained in the following sections.

8.4 Feature analysis and method acceptance testing

In order to perform feature analysis and method acceptance testing (validation 1 and 2), one group of post-test only design was used in this study. The evaluation study presented in Figure 8.3 was designed and performed.

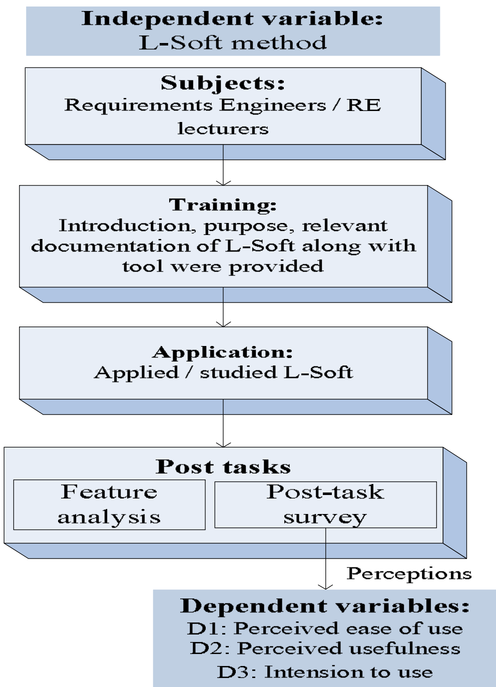


Figure 8.3: The study design

8.4.1 Independent variable

The independent variable is the method being evaluated. In this study, the independent variable is the L-Soft method.

8.4.2 Subjects

The target participants (i.e. the experts) are the experienced software engineers, the lecturers experienced in teaching RE and the researchers doing research in RE. A total of 13 experts participated voluntarily in the study in response to an invitation by email. The average experience of the participants was five years. A brief profile of the participants is presented in Table 8.2.

Table 8.2: Profile of the participants

No.	Position	Institute
01	Principal Software engineer	360 Training, Karachi, Pakistan
02	Researcher	HPI, Potsdam, Germany
03	Software Engineer/Visiting Faculty Member	ISRA University, Hyderabad, Sindh, Pakistan
04	Lecturer	Sukkur Institute of Business Administration (IBA), Sukkur, Sindh, Pakistan
05	Lecturer	University of Malaya, Malaysia
06	Assistant professor	University of Twente, Enschede, the Netherlands
07	Senior Software Engineer	Tyler Technologies, Renton, Washington, USA.
08	Research Assistant	Auckland University of Technology, New Zealand
09	Principal Software Developer	Institute of Business Administration (IBA), City Campus, Karachi, Pakistan
10	Software Developer	The Netherlands
11	Web Developer	Dell, Malaysia
12	Senior software Engineer/Researcher	Auckland University of Technology, New Zealand
13	Senior Software Engineer	FinalTier Systems, 2705 Woodley Place NW Washington DC 20008, USA.

8.4.3 Training

The introduction, purpose, relevant documentation of L-Soft and the hyperlink of the web-based tool were sent to the participants through email, and they were requested to go through the documentation and apply the method.

8.4.4 Application

The participants have studied the documents and applied the method using the L-Soft tool. They logged in as a user in the system and performed the tasks using the L-Soft method in order to assess the method. All of the tasks performed by them were recorded by the admin (the researcher).

8.4.5 Post-tasks

After the participants have studied and applied the method, they were asked to complete a post task survey by filling out an online questionnaire (Appendix D), in which they were asked to give their opinions of the method and assess the method through feature analysis.

8.4.5.1 Feature analysis

The first objective was fulfilled through feature analysis. The appropriateness of a method/tool is usually assessed in terms of the features provided by the method/tool. The specific features and characteristics included in the evaluation are based on the requirements of the user population and any organisational procurement standards. Evaluators assess the extent to which the method/tool provides the required features in a usable and effective manner based (usually) on personal opinions. This type of evaluation is referred to as feature analysis and is identified as a qualitative or subjective evaluation because it usually requires a subjective assessment of the relative importance of different features and how well a feature is implemented.

Feature analysis is an extremely flexible method of method/tool evaluations. It can be applied to any type of method or tool from Word processors to Lifecycles.(Kitchenham, 1996). Feature Analysis evaluations can be organised using qualitative screening (done by a single individual who not only determines the features to be assessed and their rating scale but also does the assessment), qualitative experiment (done by a group of potential user who are expected to try out the methods/tools on typical tasks before making their evaluations), qualitative case study (performed by someone who has used the method/tool on a real project) and qualitative survey (done by people who have had experience of using the method/tool, or have studied the method/tool). (Kitchenham, 1996)

The following processes were performed for carrying out feature analysis validation.

8.4.5.1.1 Decide the required properties or features being evaluated

The set of features was identified from the research literature according to the requirements of the problem structuring and analysis process. The criteria used for feature generation is based on the one used by (Niazi, 2002) to develop the evaluation framework for the validation of the Requirement Elicitation, Analysis and Validation Method (REAVM). Five criteria were used to generate features. First, they should be geared to the problem structuring and analysis process; second, they should clearly differentiate between the stages of the problem structuring and analysis process; third, the features should be well used and well known; fourth, it should have basic objective to improve/assess the problem structuring and analysis process and fifth, it should incorporate the objectives of this research.

(El Emam & Madhavji, 1995), (Sawyer, Sommerville, & Viller, 1997), and (Niazi, 2002) identified good practices and criteria for assessing requirements engineering activities/practices. Their studies cover the complete requirements engineering process, but because our study focuses only on the problem structuring and analysis process, it is narrower in focus. A few features of the problem structuring (problem space) stage were taken from (El Emam & Madhavji, 1995) such as examining the current system and user participation. Likewise, a few features of the problem analysis (software world) stage were taken from (Sawyer et al., 1997) such as uniquely identify each requirement and use checklists for requirements analysis. Many of the features suitable for our situation were taken from (Niazi, 2002). The rest of the features were generated from the analysis of the literature based on the objectives of the study.

The list of generated features is presented in Table 8.3. The identified features cover both stages of the problem structuring and analysis process which are problem structuring (problem space) and problem analysis (software world).

Table 8.3: The list of identified features

No.	Stages	Features
	Basic	Have sound theoretical basis
		Help beginners
1.	**Problem structuring (problem space)**	Understanding the problem
		Identifying stakeholders
		User participation
		Decomposing the problem
		Structuring the problem
		Examining the current system
		Highlighting the changes needed
		Use business concerns (goals) to derive requirements
		Use scenarios to derive requirements
2.	**Problem analysis (software world)**	Generating requirements from elements of the decomposed problem set
		Collecting requirements from multiple viewpoints
		Generating requirements at different abstraction levels
		Define potential requirements
		Uniquely identifying each requirement
		Use checklists for requirements analysis
		Prioritizing requirements
		Define standard requirements structure
		Define standard templates for requirements description

8.4.5.1.2 Scoring/ranking system that can be applied to all the features

Once the features are identified, the metrics must be defined in order to estimate the features (Grimán, Pérez, Mendoza, & Losavio, 2006). (Kitchenham, 1996) points out that there are two types of metrics: simple and compound. On this occasion, it was decided to work with compound metrics. A compound metric is a metric in which two or more simple measures are taken and combined together to form one metrics (Batra, 2010). In a compound metric, the degree of help offered by the method must be measured or judged on an assessment criteria (Kitchenham, 1996). The following assessment is made against each feature.

- High emphasis: This means that the feature is highly emphasized in the method
- Medium emphasis: This means that the feature is averagely emphasized in the method
- Low emphasis: This means that the feature is less emphasized in method
- No emphasis: This means that the feature is not covered in the method

Table 8.4 summarizes the assessment criteria along with the value given for each of them.

Table 8.4: The ordinal scale

Value given	**Description**	**Definition**
4	High emphasis	The feature is highly emphasized in the method
3	Medium emphasis	The feature is averagely emphasized in the method
2	Low emphasis	The feature is low emphasized in method
1	No emphasis	The feature is not covered in the method

8.4.5.1.3 Carry out the evaluation

The feature analysis was performed using a qualitative survey, where the participants were asked to rate the degree to which each feature is supported by the method. The feature analysis results are presented in Table 8.5.

Table 8.5: Feature analysis results

Stages	Features	High emphasis	Medium emphasis	Low emphasis	No emphasis
Basic	Have sound theoretical basis	2	8	3	
	Help beginners	6	6	1	
Problem structuring (problem space)	Understanding the problem	4	9		
	Identifying stakeholders	3	6	3	1
	User participation	2	8	3	
	Decomposing the problem	6	6	1	
	Structuring the problem	7	6		
	Examining the current system		7	5	1
	Highlighting the changes needed	1	6	5	1
	Use business concerns (goals) to derive requirements	1	9	2	1
	Use scenarios to derive requirements	4	9		
Problem analysis (software world)	Generating requirements from elements of the decomposed problem set	3	9	1	
	Collecting requirements from multiple viewpoints	2	8	2	1
	Generating requirements at different abstraction levels		6	6	1
	Define potential requirements	2	10	1	
	Uniquely identifying each requirement	4	5	4	
	Use checklists for requirements analysis	4	6	3	
	Prioritizing requirements	4	6	3	
	Define standard requirement structure	1	9	3	
	Define standard templates for requirements description	2	8	3	

8.4.5.1.4 Analyse and interpret the results

To calculate the mean of the responses, the mean values of features of each stage were calculated for each respondent (table 8.6). Then the mean values of all respondents were calculated for all three stages using the formula:

$$\bar{X} = \frac{\sum x}{N} = \frac{respondent\ 1 + respondent\ 2 + \cdots . + respondent\ 3}{total\ number\ of\ respondent}$$

Table 8.6 presents the mean values of each stage for each respondent and finally the mean values of all respondents for the three stages.

Table 8.6 Calculation of mean values for three stages of features

	Basic feature	**problem structuring features**	**problem analysis features**
Respondent 1	3.00	2.89	2.89
Respondent 2	3.00	2.78	2.33
Respondent 3	3.00	2.89	2.56
Respondent 4	3.00	3.11	3.44
Respondent 5	3.00	2.89	2.78
Respondent 6	3.50	3.00	2.44
Respondent 7	3.00	3.00	3.00
Respondent 8	3.50	3.00	3.00
Respondent 9	3.50	3.22	3.00
Respondent 10	4.00	3.56	3.67
Respondent 11	3.00	3.00	3.00
Respondent 12	3.00	2.78	3.11
Respondent 13	2.50	3.00	2.89
Mean value $\left(\frac{\sum x}{N}\right)$	3.154	3.009	2.932

The chart showing mean values of the feature analysis results for the L-Soft method are presented in Figure 8.4.

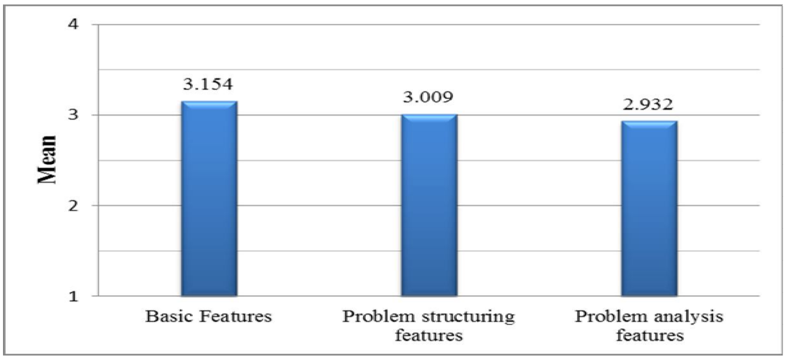


Figure 8.4: The mean values of the feature analysis results

The mean values of the feature analysis results (Fig. 8.4) show that the L-Soft method supports all features to a medium-emphasis degree according to experts' opinions. Therefore, it can be concluded that the L-Soft method is appropriate to be used for performing the problem structuring and analysis process with medium emphasis.

However, in assessing whether the method fits the needs, a Shapiro-Wilk test was conducted. It involves arraying the sample values by size and measuring the fit against expected means, variances and co-variances. The hypotheses involved in this test are:

H_o: The data is fit

H_1: The data is not fit

The null hypothesis for this test is the data is fit. The Prob < W value listed in the output is the p-value. If the chosen alpha level is 0.05 and the p-value is less than 0.05, then the null hypothesis that the data is fit is rejected. If the p-value is greater than 0.05, then the null hypothesis is not rejected. The output is presented in Table 8.7.

Table 8.7: The Shapiro-Wilk test

	Shapiro-Wilk		
	Statistic	df	Significance value
Basic features	0.817	13	0.521
Problem structuring features	0.835	13	0.608
Problem analysis features	0.946	13	0.535

The results of the Shapiro-Wilk test (Table 8.7) show that the p-value of all stages is greater than 0.05. Therefore, we do not have enough evidence to reject a null hypothesis. Based on that, it can be seen that the data for all the three stages, that are basic, problem structuring and problem analysis features, is fit. Therefore overall, we can conclude that the L-Soft method is appropriate to be used for performing the problem structuring and analysis process.

8.4.5.2 Method acceptance testing

In order to fulfil the second objective, the following hypothesis was set.

Hypothesis: L-Soft is a light weight method that can be used to perform the problem structuring and analysis process and can be adopted in teaching RE course in universities.

The hypothesis is divided in to following three items and can be considered true if these items are met:

- L-Soft is a light-weight (easy to understand and apply) method.
- The L-Soft method is useful for performing and teaching the process of problem structuring and analysis.
- The L-Soft method can be successfully adopted in teaching RE course in universities.

These three items of the hypothesis are tested using three perception-based measures of MEM presented in the following section.

8.4.5.2.1 Perception-based dependent variables

Three perception based variables were used to evaluate the method.

D1: Perceived ease of use. This was measured using six items on the post-task survey (Questions 1, 4, 6, 9, 11 and 14).

The items used to operationalize perceived ease of use were adapted from (Davis et. al, 1989) and (D.L. Moody, 2003)'s studies, with changes in wording to fit the use of a method as opposed to the use of a computer system. A total of six items were used to measure the perceived ease of use as listed below. Note that half of the items in the survey were negated to avoid monotonous responses.

Q1. I found the procedure of applying the method complex and difficult.

Q4. Overall, I found that the tool based on this method is difficult to be used.

Q6. I found that the method is easy to be understood

Q9. I found that it is difficult to apply the method for the given problem

Q11. I found that the given guidelines are clear and easy to understand

Q14. I am not confident that I am now competent to apply and teach this method in RE course.

D2: Perceived Usefulness. This was measured using eight items on the post-task survey (Questions 2, 3, 5, 7, 8, 12, 13, and 15). The items used to operationalize perceived usefulness were also adapted from (Davis et. al, 1989) and (D.L. Moody, 2003)'s studies, with changes in wording to fit the use of a method as opposed to the use of a computer system, and to reflect the objectives of the method (as usefulness is defined in terms of how the method achieves its objectives).

Q2. I found that this method would reduce the effort required to analyse and structure customer's problem into set of requirements.

Q3. The requirements extraction process using this method is more difficult for students to understand

Q5. This method would make it easier for lecturers/users to verify the extracted requirements

Q7. Overall, I found the method to be useful in teaching and learning problem structuring and analysis process

Q8. By using this method, it is more difficult to analyse and structure the customer's problem

Q12. Overall, I think this method does not provide an effective solution to the problem of teaching problem structuring and analysis

Q13. Overall, I think this method is an improvement to the standard requirements analysis methods taught in universities

Q15. By using the software tool it is easier to understand and apply the method

D3: Intention to Use. This was measured using two items on the post-task survey (Q10 and Q16). Statements used to operationalize intention to use were also adapted from (Davis et. al, 1989) and (D.L. Moody, 2003)'s studies. The intention to use is operationalized using two items:

> Q10. I would definitely not use this method to perform/teach problem structuring and analysis process in requirements engineering.
>
> Q16. I intend to use this method in preference to the standard requirements analysis methods if I have to work/teach to extract the requirements from customer's problem in the future.

The items defined for perceived ease of use, perceived usefulness and intention to use were combined together into a post-task survey consisting of 16 items. Each item was measured using a 5 point Likert scale. To ensure the balance of items in the questionnaire, half of the statements were negated to invite the attention of respondents who might become increasingly alert to manipulated question items (Hu & Chau, 1999). In addition, items were arranged in a random order to reduce the potential ceiling effect that could induce monotonous responses to question items measuring the same construct (Hu & Chau, 1999).

No performance based data was collected as part of this experiment as this is only meaningful in comparison between methods.

8.4.5.2.2 Validity and reliability analysis

To evaluate the results for perceived ease of use, perceived usefulness and intention to use, it is first necessary to evaluate the validity and reliability of their empirical indicators.

Construct validity

Construct validity refers to establishing correct operational measures for the concepts being studied (Kitchenham, 1996).

For evaluating construct validity, factor analysis is the preferred technique among researchers (D.L. Moody, 2003). However in this study, because the sample size was too small (only 13), an inter-item correlation analysis was used to evaluate the validity of dependent variables. Inter-item correlation is more commonly used in psychometric and risk assessment tool research and is seen by many to be the most important index of test reliability. The test is used to see if any of the questions ("items") do not have responses that vary in line with those for other tests across the population. (Kline, 2000)

Two items failed the validity test: Q11 (Perceived ease of use) and Q12 (Perceived usefulness) as they were found to have low convergent validity. Therefore, both items were removed from the analysis.

Reliability Analysis

Reliability refers to demonstrating that the study can be repeated with the same results (Kitchenham, 1996). Reliability analysis was conducted on the items used to measure perceived ease of use (excluding Q11), perceived usefulness (excluding Q12) and intention to use, using a coefficient (Cronbach's alpha) method. Reliability coefficients usually greater than the recommended level of 0.7 are considered acceptable and the factor structure replicated well when used in different settings.

Table 8.8 shows the reliability test results of the perceived ease of use, perceived usefulness, as well as intention to use among experts. The reliability test resulted in Alpha values of 0.7 or above indicating that the results are reliable and valid.

Table 8.8: Item Reliabilities for Dependent Variables

Dependent Variables	**No of items**	**Cronbach Alpha**
Perceived ease of use	5	0.710
Perceived usefulness	7	0.720
Intention to use	2	0.700

8.4.5.2.3 Likelihood of adoption in practice

One-sample t-tests were conducted on the values of each construct to determine whether they were significantly different to the zero point of the scale (3). It is a hypothesis test for answering questions about the mean where the data is a random sample of independent observations (Easton & McColl). It also allows us to test whether a sample's mean (of a normally distributed interval variable) significantly differs from a hypothesized value.

Table 8.9: The results of statistical comparisons

Dependent Variables	**Mean**	**Standard Deviation**	**Significance value**
Perceived ease of use	3.58	0.4672	0.000
Perceived usefulness	3.82	0.3438	0.000
Intention to use	3.73	0.4385	0.000

The results (Table 8.9) show that the mean of perceived ease of use is 3.58, the mean of perceived usefulness is 3.82 and the mean of intention to use is 3.73. It is obvious that the respondents had good perceptions towards the L-Soft method since the mean for each variable is quite high. Referring to the positive significance value (two-tailed), it suggests that the L-Soft method has a high likelihood of being adopted in practice.

From high value of perceived ease of use of L-Soft method, it can be concluded that "L-Soft is a light-weight (easy to understand and apply) method"; from high value of perceived usefulness of L-Soft, it can be concluded that "The L-Soft method is useful for performing and teaching the process of problem structuring and analysis"; and from high value of perceived intention to use, it can be concluded that "L-Soft method can be successfully adopted in teaching RE course in universities", proving the three items of the hypothesis true.

Therefore the hypothesis of the study, “L-Soft is a light weight method that can be used to perform the problem structuring and analysis process and can be adopted in teaching RE course in universities” can be considered true.

8.4.5.2.4 Deficiencies in L-Soft

At the end of the post-task survey, the participants were asked their opinions about L-Soft and to select aspects of the L-Soft method that they found to be deficient and elaborate on their responses including suggesting ways to improve the method.

A few positive comments on the method and tool are:

- “I like the idea to structure the process for students because as far as I know, there is usually no 'standard process' to be taught.”
- “Overall I found the tool easy to use and comprehensively elaborated. It was a nice experience to practice with such a simple tool.”
- “This tool will definitely be helpful for novice software analysts and software engineering students to understand the requirements engineering process. Such a method provides a level of guidelines that can guide users in a structured way from gathering requirement until arriving to a set of software requirements that is easy-to-understand. “
- “From my personal point of view, such a tool will be helpful to be used in teaching purposes and can help software engineering students to understand how requirements can be collected and extracted.”

In addition, the participants also pointed out many deficiencies and gave their suggestions on improving different aspects of the method. Their responses on the deficiencies of several aspects of L-Soft and the suggestions for improvement are presented in Table 8.10.

Table 8.10: The deficiencies in the L-Soft method and the suggestions for improvement

No.	**Deficiency in**	**Frequencies**	**Elaborations/suggestions**
1)	The steps of the method	38%	- There is more to gain in explicitly defining different stakeholders and tying them to work-processes, features and requirements. - The input is quite informal, which complicates the re-use of already covered information.
2)	The concepts used for the method	23%	- No elaborations or suggestions
3)	The procedure used to apply the method	15%	- The approach to analyse existing and new scenarios need to be improved. There should be a more systematic way to help the requirements engineers to refine the scenarios using the new/upgraded system. - For the existing work processes and refined work processes, the important part should be to help them to refine the process. - The classification of requirements priority in High/Medium/Low levels usually doesn't work; the majority of the requirements have high priority.
4)	The guidelines provided with the method	31%	- I am not 100% sure about the explanation of each step. I have to keep referring to the solved Case Studies to confirm whether I have done correctly and found I had misunderstood some steps in Problem Recognition, Problem Understanding, and Problem Organization. - I like the method but I have to go through the case study to understand it, that’s why I would like you to improve the guidelines and demo.

Table 8.10, continued

5)	The L-Soft tool	15%	- Add a view where the requirements are sorted by priority, with the possibility to move them up and down. - Allow users to enter the goals/benefits line by line, instead of one paragraph. - It should be allowed to change the sequence of the requirements. - The tool may provide a function to help users generate Requirements Specifications for projects. - Instead of asking the users to upload use case diagram, they should be shown the old scenario when they enter the new scenario. That will help the requirements engineer to refine the work scenario by comparing it with the older versions. I think it is good to have the step-by-step of the refined work processes in texts first before drawing the Use Case Diagram.
6)	The solved case studies	8%	- No elaborations or suggestions given
7)	The getting started demo	31%	- The demo is good, but will be better if it can show the title of the step/feature before showing the demo of a particular step, e.g. First Step: Create new project. - It can be a simulation based help for users, there are software out there (e.g. VIEWLET) to create interactive helps.

A few more suggestions for extension of the work are:

- "Add more requirements properties as present in the Volere Template, elaborate more on stakeholder definition e.g. using Ian Alexander's Stakeholder Onion to ensure that all important groups are covered, and use more structured scenarios."

- “However, the method/tool used here emphasizes more on problems abstraction and decomposition than on the requirement management process itself. Some important requirement engineering elements, such as requirement categorization, change possibility and classification scheme used to classify requirements, seem to be missing from the tool. This will make the tool hard to use in an industry setting, where there is usually a heavy use of the above mentioned elements.”

Before moving to the next validation method, a few changes were made (especially in the L-Soft tool) based on the participant’s suggestions in order to overcome the deficiencies found.

8.5 The formal experiment

To fulfil the third objective, the research called for a formal experiment with undergraduate software engineering students. Recall that the third objective is to evaluate whether the L-Soft method is a light-weight method that can enable students to perform the problem structuring and analysis process and can be adopted in teaching students as part of the RE course in universities.

To discover whether the L-soft method does provide advantages to students in universities, the evaluation was organized as a formal experiment where many subjects (i.e. students) were trained using methods that are being evaluated and were asked to perform a task (or variety of tasks). The subjects were assigned to the method such that the results are unbiased and can be analysed using standard statistical techniques. The controlled experiment was conducted in the class rooms of Faculty of Computer Science & Information Technology (FCSIT), University of Malaya (UM).

The formal experiment approach is selected because it is likely to give the most trustworthy results (Kitchenham, 1996). Another key motivation for using a formal experiment rather than a case study is that the results of the experiments are usually more generalizable than those of a case study (Pfleeger, 1995).

The method evaluation model was used to evaluate the comparative efficacy of a two methods for performing the problem structuring and analysis process.

8.5.1 Experimental study design

This is the post-test only control group design experiment. This type of design involves administrating a treatment (i.e. L-Soft method) to one type of groups (treatment groups), with observations taken only at the end following the treatment. These observations are contrasted against the control groups that were trained using an existing analysis method. It is from this contrast that the results of this study are drawn. The design of experimental study is shown in Figure 8.5.

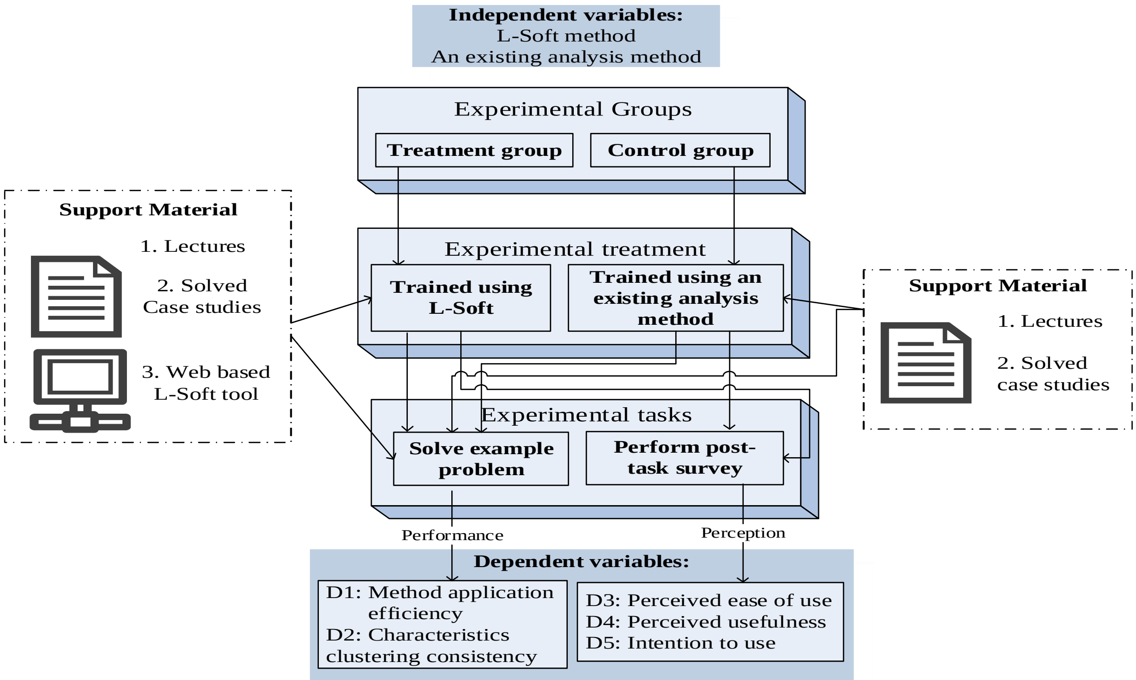


Figure 8.5: The Experimental study design

8.5.2 Independent variables

The independent variable is the variable that is assumed to be the cause of the effect. It is the variable that the researcher varies or manipulates in a specific way in order to learn its impact on the outcome variable. (Johnson & Christensen, 2003)

In this experiment, the independent variables are the methods being evaluated and compared that are:

- L-Soft method
- An existing analysis method

The L-Soft method is the method proposed in this research (chapter 6). While an existing analysis method is the requirements analysis method normally taught to students in the classrooms for requirements analysis and extraction (especially in our institution). The analysis method uses different techniques for requirements extraction, the steps followed by the method include identification of scope of the project (through context diagram), project goals, stakeholders, business events, use cases, and finally requirements. The detail of the process can be found in (Robertson & Robertson, 2012) and (Bray, 2002).

8.5.3 Experimental groups

A total of 36 students participated in this experiment, all of whom were undergraduate software engineering students who have taken and passed RE course. The participants were randomly assigned into groups of two, making a total of 18 groups. The groups were then randomly divided into two types: treatment groups and control groups.

8.5.4 Experimental treatment

The participants were given the theory of RE before the training of the particular method. The treatment groups were trained with the L-Soft method, while the control groups were trained using an existing analysis method. The training time was around two hours. The support materials used for training were lectures, solved case studies and web-based tool (in the case of the L-Soft method).

8.5.5 Experimental tasks

Each of the 18 groups was given the same example problem i.e. Practical Online Help Desk (OHD) for campus facilities and were asked to analyse and structure the problem and generate the requirements using the method that they had learnt. This example problem was chosen since it constituted a sizable and complex problem that would still be feasible within the constraints of the university course. Both types of groups were given customer's initial needs for the example problem. They were given one hour to perform the assigned tasks. The data collected were the set of requirements generated by all the groups. Finally, they were asked to complete a post-task survey, in which they were asked to give their perceptions on the method that they have used.

8.5.6 Dependent variables

The dependent variable is the effect or outcome variable (Johnson & Christensen, 2003). The dependant variables determine whether a method is successful or not (D.L. Moody, 2003). Two types of dependent variables are being used in order to measure the successfulness of a method, which are performance-based dependent variables and perception-based dependent variables.

8.5.6.1 Performance-based measures

Two performance based dependent variables were used to evaluate the methods.

D1: Method application efficiency. This construct was measured by the number of requirements produced by the subjects by solving the example problem using the method they have learnt. This provides a measure of actual efficiency.

D2: Characteristics clustering consistency. This was measured by rating the characteristics of the requirements produced by each group. This provides a measure of consistency between different people using the same method.

The requirements characteristics are those properties that are expected to be present in the requirements extracted using the method being evaluated. Unit of analysis are individual requirements, and we are interested in the differences in the requirements characteristics. The characteristics against which each requirement is rated are presented in Table 8.11.

Table 8.11: Requirements characteristics

No.	Characteristic	Description
1	Clear, concise & unambiguous	The degree to which requirement is easily read and understood by nontechnical people, unambiguous (should have one interpretation).
2	Necessary	The degree to which the requirement must be present to meet system objectives and customer really need it.
3	User's needs	The degree to which the requirement is focused on the needs of users.
4	Feasible	The degree to which the requirement is possible to implement within known capabilities and limitations.
5	Level of detail	The degree to which the requirement expresses one thought and written in appropriate level of detail.
Scale for characteristics: These were measures using a 5-point Likert scale. The scale is the same for all characteristics and is defined as: 5- Very high, 4- High, 3- Neither high nor low, 2- Low, 1- Very low.		

8.5.6.1.1 Data collection

The source of data is the requirements ratings for the requirements characteristics (defined in Table 8.5) of the requirements generated by the experimental groups. The ratings for each characteristic were assigned to two external researchers who independently performed the ratings. The external researchers checked 18 sets of requirements generated by 18 groups. This process involved examining the title and description of each requirement and giving a rating on the appropriate scale of each and every characteristic. All ratings were conducted blindly, i.e. without knowledge of which group authored which requirements.

The researchers chosen for the rating procedure have more than 3 years of experience in RE. One of them was a PhD candidate, while the other was a PhD holder. The requirements rating data collection instrument was designed for this purpose. The instrument was administrated on each requirements evaluator to collect the requirements rating data. The instrument was operationalized through an MS Excel spreadsheet file, and the organization of the spreadsheet is organized from the structure of Table 8.12.

Table 8.12: Requirements ratings data entry template

Req. ID	**Caption**	**Description**	Clear, concise & unambiguous	Necessary	Focus on User's needs	Level of detail
R001						
R002						

Each requirement takes one row of this table. The first three columns are where the information pertaining to the requirement is given to the evaluators. Specifically, for each requirement, there are three pieces of the information given to the evaluator: Requirement Id, a caption and a description. The requirement ID is a numerical value that uniquely identifies the requirement. The caption indicates the title given to the requirement. The description is the requirement itself. The next four columns are where the evaluators enter the ratings for the particular requirements characteristic given in the column header. The evaluators filled out this part of the instrument with reference to the list of requirements characteristics, their definitions, and the scales to use for each characteristic (defined in Table 8.8). In order to remove possible bias during the rating process, the table does not contain any information that can associate any given requirements with the specific teams that generated the requirements. The results of each individual evaluator's assessment are merged into another MS Excel sheet which is organized based on the structure of Table 8.13.

Table 8.13: Sampling rating of requirements

Req. ID	Clear, concise & unambiguous	Necessary	Focus on User's needs	Level of detail
R001	4, 4, *4*	5, 4, *4.5*	4, 3, *3.5*	4, 3, *3.5*
R002	5, 4, *4.5*	5, 5, *5*	5, 5, *5*	4, 4, *4*
R003	3, 3, *3*	3, 2, *2.5*	3, 4, *3.5*	4, 3, *3.5*

Table 8.13 shows the ratings of three example requirements, R001, R002 and R003.

R001. The system shall allow users (students, faculty, lab-assistants and others) to register into the system by entering their personal information.

R002. The system shall allow the users to login to the system if they supply the correct username and password.

R003. The system shall ensure that unauthorized users cannot access the system.

The first two numbers in each column represent ratings given to the requirement for the given characteristic by the two independent evaluators (the rating scale is explained in Table 8.8). The third value in italic is the final agreed upon value.

8.5.6.1.2 Threats to validity

The threats are classified into those internal and external to project, as well as construct and conclusion validity. The threats that are considered relevant to this study are focused here.

i) Internal validity

Internal validity deals with whether we can infer that a relationship between two variables is casual, and not due to any confounding factors (Johnson & Christensen, 2003). A few types of internal validity threats, relevant to our study (based on multigroup design) are discussed in this section.

Differential selection: This is when possible characteristics of the subjects may, by chance, differ between the two types of groups and possibly affect the quality of the data (Ferrari, Miller, & Madhavji, 2010). In our study, such a characteristic is the participant's RE educational knowledge and software engineering (SE) backgrounds; participants with different knowledge and background could possibly perform differently in the projects. To identify any such possible outlier participants, prior to study, each participant was communicated about their SE background and RE knowledge so that any subjects with non-SE background or not sufficient RE knowledge could be identified, but none had any such background. This coupled with the knowledge that every participant was a full-time software engineering student (mostly final year) and had taken and passed RE course recently or one semester back. As RE course is only offered to students in their fourth and fifth semester and has many prerequisites so students can only take this course in their third or final years. This ensured that they had similar knowledge and backgrounds.

Differential mortality: This occurs when a physical or mental change occurs to participants during study that is not "equal" between the two types of study groups (Ferrari et al., 2010). This threat cannot exist in our study because of the small duration of the experiment (around 3 hours). However if the experiment is being performed during the full semester course, the threat may exist and researchers need to consistently review and asses participant's submissions and performance to avoid the effects of differential mortality threats.

Researcher's bias: This occurs when the researcher, knowingly or unknowingly, influences the outcome of the study (Ferrari et al., 2010). This threat exists in this study because of the subjective nature of the requirements characteristics ratings, also the experimental sessions were handled by the author who acted as a facilitator. To mitigate this threat, the author's participation was controlled.

All studies managed by a facilitator were managed consistently (i.e. the same level of help or non-help was provided to all participants). Also multiple researchers and domain experts participated in the study. Additionally, the results were cross-validated.

ii) External validity

External validity refers to the degree to which the results of a study can be generalized across a population, time or place (Johnson & Christensen, 2003). Population validity can exist when generalizing to the academic institutions in other parts of the world. This severely threats the external validity of study. The reason of using students of this university in our study was the availability of resources in the parent institution, as it would have been extremely difficult to conduct this first time controlled study in other institutions. Because of the factors that the RE course is being taught using standard topics and due to the nature of RE, students usually face problems in learning RE concepts. Therefore, it is expected that a replication of this study in a different site and/or with different size teams shall generate the same results. The use of students should not diminish the results of this study, as important results have been found in other students based SE studies such as in (Ferrari et al., 2010) and (Memon, Ahmad, & Salim, 2013). This preliminary experimental study, therefore, can provide the groundwork for future studies of method validation in wider contexts.

iii) Construct validity

Construct validity refers to the extent to which a measurement corresponds to theoretical concepts (i.e., constructs) concerning the phenomenon under study (Johnson & Christensen, 2003). In this study, the constructs were the requirements characteristics. These were measured by an instrument created and used by external researchers (see sect. 8.2.6). The constructs and the instrument were already validated as they were taken from existing literature with little modification such as from (Ferrari et al., 2010).

iv) Conclusion validity

Conclusion validity is the degree to which conclusions we make based on our findings are reasonable (Johnson & Christensen, 2003). There are three ways in which conclusion validity can be improved: statistical power, reliability, and proper implementation of study methods (Ferrari et al., 2010). In this study, statistical power is not an issue, as statistical tests are performed on ratings from the generated requirements which were elicited by the two types of teams. Also, the study design and statistical tests (see Sect. 8.5.6.1.4) accounted for the difference between the unit of analysis (teams) and unit of observation (requirements). Also two different researchers were used to rate each requirement in order to achieve a reliability of the rankings.

8.5.6.1.3 Control of confounding variables

An extraneous variable is a variable that may compete with the independent variable in explaining the outcome of a study. A confounding variable is an extraneous variable that does cause a problem (influences independent and dependent variable) because it does have a relationship with the independent and dependent variables. Variety of techniques is used to control confounding variables that eliminate the differential influence an extraneous variable may have for the comparison groups in a research study. Differential influence occurs when the influence of an extraneous variable is different for the various comparison groups. If the comparison groups are the same on all extraneous variables at the start of experiment, then differential influence is unlikely to occur. The control techniques are essentially attempts to make the groups similar or equivalent. The only systematic difference between the groups should be the variation of the independent variable. (Johnson & Christensen, 2003)

In this experiment, the confounding variables were controlled using random assignment and matching.

Random Assignment: The purpose of random assignment is to take a sample and randomly divide it into two or more groups that represent each other. Thus, the problem of differential influence is controlled by insuring that each participant has an equal chance of being assigned to each comparison group. (Johnson & Christensen, 2003) In this experiment, the participants (students) as well as the characteristics they bring with them were equally likely to be assigned to each comparison group. Therefore the research participants and their characteristics were distributed approximately equally in all comparison groups creating equivalent groups.

Matching: Matching controls for confounding extraneous variables by equating the comparison groups on one or more variables that are correlated with the dependent variable. It eliminates any differential influence of the matching variables. In our case, the matching variable was student's SE background and basic RE knowledge they possess. Both comparison groups were matched on these variables.

8.5.6.1.4 Data analysis results and interpretations

The produced requirements were initially checked by the facilitator and any wrong, duplicated and meaningless requirement was removed. After initial check, the treatment group (trained using L-Soft) collectively produced 193 requirements, whereas the control group (trained using an existing analysis process) collectively produced 180 requirements. Therefore in terms of method application efficiency, L-Soft can be considered more efficient than the existing analysis process.

However, we are primarily interested in exploring whether there are significant differences in the characteristics of the requirements generated by both teams (experimental vs. control) and not simply the teams themselves. Specifically, we conducted a separate statistical analysis that incorporated the possible effect of the different teams on the requirements characteristics.

A Two-way ANOVA was conducted that incorporated the possible effect of the different teams on the characteristic ratings.

Table 8.14: The mean scores of the two types of groups for selected characteristics

	Existing method	**L-soft method**
	Mean	Mean
Clear, concise & unambiguous	3.71	3.83
Necessary	3.76	3.93
User's needs	3.74	3.91
Level of detail	3.39	3.62

Referring to the results in Table 8.14, it can be observed that on average, the L-Soft method scored higher on all of the requirement characteristics which are clear, concise & unambiguous, necessary, user's needs as well as level of detail.

According to Table 8.15, it appears that there is a significant main effect of both the L-Soft and existing method on all of the requirement characteristics due to the significance level of less than 0.05.

Table 8.15: Multivariate Tests

Effect	Value	F	Hypothesis df	Error df	Sig.
Method Pillai's Trace	0.034	5.821[b]	4.000	661.000	0.000

The results of the analysis of Table 8.15 found that overall, there are main effects for both the L-Soft and existing method in terms of all requirement characteristics. Therefore, the main effects of each variable can be identified in Table 8.16. Based on the output, it appears that there is a main effect of methods in all of the requirement characteristics. Therefore, it can be concluded that the method affects the characteristic of clear, concise & unambiguous, necessary, user's needs as well as level of detail.

Table 8.16: Tests of Between-Subjects Effects

Source Dependent Variable	Type III Sum of Squares	df	Mean Square	F	Significance value
Clear, concise & unambiguous	2.347[a]	1	2.347	5.812	0.016
Necessary	4.995[b]	1	4.995	13.927	0.000
User's needs	4.782[c]	1	4.782	12.664	0.000
Level of detail	8.269[d]	1	8.269	18.674	0.000

a. $R^2 = 0.009$ (Adjusted $R^2 = 0.007$)

b. $R^2 = 0.021$ (Adjusted $R^2 = 0.019$)

c. $R^2 = 0.019$ (Adjusted $R^2 = 0.017$)

d. $R^2 = 0.027$ ($R^2 = 0.026$)

The R^2 value below Table 8.15 shows that the method only accounted for 0.9 percent for clear, concise & unambiguous, 2.1 percent for necessary, 1.9 percent for user's needs as well as 2.7 percent for level of detail.

Table 8.17: Pairwise comparison test

Dependent Variable	Mean Difference (L-soft vs. Existing)	Std. Error	Significance value
Clear, concise & unambiguous	0.121	0.050	0.016
Necessary	0.176	0.047	0.000
User's needs	0.172	0.048	0.000
Level of detail	0.226	0.052	0.000

Based on the results shown in Table 8.17, it clearly shows that the L-soft method is better than the existing method in terms of all of the requirement characteristics since the values for all of the variables are less than 0.05. Overall, we can conclude that the L-Soft method can be adopted in teaching to students as part of the RE course in universities.

8.5.6.2 Perception-based measures

Three perception based variables were used to evaluate both methods.

D3: Perceived Ease of Use.

D4: Perceived Usefulness.

D5: Intention to Use.

Two measurement instruments were used for L-Soft group (Appendix E) and existing analysis method group (Appendix F). Both instruments were almost same (post task survey) as in the previous study described in Section 8.4 (Appendix D).

The Actual Usage i.e. the extent to which a method is used in practice (D.L. Moody, 2003) was not evaluated as part of this study, as this was not possible in an experimental context. However, Davis et al (1989) found that the intentions after only a one hour introduction to a computer package predicted the usage behaviour 14 weeks later. In this experiment, participants spent two hours learning and using the methods, so we can argue that the intentions should predict the future usage reasonably well.

8.5.6.2.1 Validity and reliability analysis

Construct Validity

An inter-correlation analysis was carried out since the sample size was too small in this experiment. No question was found to have low convergent validity and therefore, no question was removed from the analysis.

Reliability

A reliability analysis was conducted on the items used to measure perceived ease of use, perceived usefulness and intension to use using the coefficient (Cronbach's alpha) method. Reliability coefficients usually greater than the recommended level of 0.7 is considered acceptable and the factor structure replicated well when used in different settings.

Table 8.18 shows the reliability test of the perceived ease of use, perceived usefulness, as well as intension to use among students. That reliability test resulted with the Alpha value that is 0.7 or above indicating that the results were reliable and valid.

Table 8.18: Item Reliabilities for Dependent Variables

Dependent Variables	**No of items**	**Cronbach Alpha**
Perceived ease of use	6	0.773
Perceived usefulness	8	0.731
Intention to use	2	0.710

8.5.6.2.2 Comparison of methods

Table 8.19 summarises the results for both experimental groups on all dependent variables.

Table 8.19: Comparison of Experimental Groups

	Experimental Groups	
Dependent Variables	**L-Soft Method**	**Existing Analysis Method**
Perceived ease of use	3.87	3.13
Perceived usefulness	4.08	3.25
Intention to use	4.19	3.44

All of the comparison groups yielded significant results. The L-Soft method performed significantly better than the existing analysis method on all three dependent variables (i.e. perceived ease of use, perceived usefulness and intension to use).

8.5.6.2.3 Likelihood of adoption in practice

One-sample t-tests were conducted on the values of perceived ease of use, perceived usefulness and intention to use. It is a hypothesis test for answering questions about the mean where the data are a random sample of independent observations.

It also allows us to test whether a sample mean (of a normally distributed interval variable) significantly differs from a hypothesized value.

Table 8.20: Results of Statistic Comparisons

Dependent Variables	**L-Soft Method**	**Existing Method**
Perceived ease of use	Yes (0.000)	Neutral (0.000)
Perceived usefulness	Yes (0.000)	Neutral (0.000)
Intention to use	Yes (0.000)	Neutral (0.000)

Mean response levels indicate the strength of response to an individual question. Strongly-held opinions are represented by either a 1 on the negative side or by a 5 on the positive side of the scale. The midpoint of the scale is 3, therefore when the responses are averaged, an average or neutral response would be 3.00. Mean response levels above 3.00 suggest a positive overall response while those below 3.00 suggest a negative overall response. Mean response levels of, say, 2.50 or 3.50 can be considered substantially negative or positive opinions. Mean response levels can be considered exceptionally low or high if they approach values of, say, 2.00 or 4.00, respectively.

The likelihood of adoption in practice was evaluated by comparing the values of perceived ease of use, perceived usefulness and intention to use for each group. Table 8.20 summarises the results of the one-sample t-tests. All comparisons were found to be significantly positive for all variables in the L-Soft method which suggests that it is highly likely to be adopted in practice. For the existing method, all comparisons were found to be significantly neutral.

8.6 Comparison of the formal experiment results of the L-Soft group to the method acceptance testing results

Significance tests were carried out between the experts' perception and the students' perception towards the L-Soft method.

Based on the result obtained from Table 8.21, only one comparison is significant which is the intention to use variable since its p-value is less than 0.05.

Table 8.21: Differences between the Experts' Perception and the Students' Perception towards the L-Soft Method

Construct	Expert's Perception	Student's Perception	Significance Value
Perceived ease of use	3.58	3.87	0.164
Perceived usefulness	3.82	4.08	0.120
Intention to use	3.73	4.19	0.021

Referring to the results of Table 8.21, we can conclude that the students have more intention to use the L-Soft method rather than the experts. However, no difference was found between the two studies in the respondents' perceived ease of use and perceived usefulness.

8.7 Discussion

Overall, the evaluation of the L-Soft method showed positive results. The three objectives of the study were fulfilled by performing the selected validation methods and the hypothesis were tested through statistical evaluation of the results. The response to each research objective is discussed in terms of the evaluation results.

> *Objective 1: To evaluate that* the *L-Soft method is appropriate to be used for performing* the *problem structuring and analysis process.*

The appropriateness of the L-Soft method that is, how well it fits the needs of performing the problem structuring and analysis process, was evaluated through feature analysis. The features that a problem structuring and analysis method should support were identified and software engineers and RE lecturers were asked to rate the degree to which the identified features are supported by the L-Soft method. The mean scores of the results showed that L-Soft supports all features with medium emphasis.

The Shapiro-Wilk test was then conducted to assess whether the L-Soft method fit its needs and the overall results showed that the method is appropriate for performing the problem structuring and analysis process, hence objective 1 was achieved.

> *Objective 2: To evaluate that L-Soft can be used to perform* the *problem structuring and analysis process and can be adopted in teaching students as part of* the *RE course in universities, and to highlight the areas where L-Soft has deficiencies.*

The perception-based measures of the method evaluation model (Figure 8.1) were used to measure the perceived adoption in practice of the L-Soft method. Software engineers and RE lecturers studied and applied the L-Soft method and responded to a post-task survey. The validity and reliability of all the constructs used to measure the perceived ease of use, perceived usefulness and intention to use were conducted through inter-item correlation analysis and Cronbach Alpha analysis. The responses were then measured using one-sample t-tests and the results suggested that the L-Soft method has a high likelihood of being adopted in practice for teaching students the problem structuring and analysis process in RE course in universities. The deficiencies found by the respondents in the L-Soft method were highlighted and their suggestions were noted in order to improve the method. Hence, objective 2 was achieved.

> *Objective 3: To evaluate that* the *L-Soft method is a light-weight method that can enable students to perform* the *problem structuring and analysis process and can be adopted in teaching students as part of* the *RE course in universities.*

A formal experiment on undergraduate software engineering students was conducted in order to verify the successfulness of the L-Soft method. A method evaluation model (Figure 8.1) was used as a basis to perform the validation. The L-Soft method was compared to an existing analysis method by training one group of students using the L-Soft method and another group of students using the existing analysis method.

The students were then assigned an example problem to solve using the method that they have learnt and generate the requirements. The generated requirements were rated by two external researchers and a two-way ANOVA was conducted in order to analyse the results. The results showed that L-Soft is more efficient and effective than the existing analysis method. The students were asked to complete a post-task in order to measure their perceived adoption in practice in terms of perceived ease of use, perceived usefulness and perceived adoption in practice. The validity and reliability of the constructs were conducted through an inter-item correlation analysis and Cronbach Alpha analysis. The likelihood of adoption in practice was evaluated by comparing the values of the perceived ease of use, perceived usefulness and intention to use for both methods. Through one-sample t-tests, all comparisons were found to be significantly positive for all variables in the L-Soft method which suggests that it is highly likely to be adopted in practice than the existing analysis method. The method was hence proven to be successful in achieving its objectives.

Finally, the students' perceptions of the L-Soft method were compared to that of the experts (i.e. software engineers and RE lecturers), and it was found that the students have more intention to use the L-Soft method rather than the experts. However, they have similar perceptions towards the method's perceived ease of use and perceived usefulness. It can therefore be concluded that the students as well as the experts found L-Soft an easy to use and useful method but students are more likely to use it in practice rather than the experts.

8.8 Tying to the research

This section shows how the research gaps identified in earlier chapters are addressed by the validation process.

In Chapter 1, the second research objective was 'to address the selected problem by formulating a method suitable to be taught to undergraduate software engineering students in RE course'. The selected problem that motivated the research was the problem of teaching problem structuring and analysis in RE (one of the major problems in REE that has yet to be investigated). The literature survey on the selected problem area in Chapter 5 highlighted that the problem can be addressed by the proposal of a new method that should fill the following three gaps:

- The method should cover the two stages of the problem structuring and analysis process and can bridge the gap between these two stages.
- The method should be a teaching method that can help students in learning and performing the problem structuring and analysis process.
- The method should be light-weight to enable students to understand and apply the problem structuring and analysis process

Firstly, feature analysis was performed and all the features that a problem structuring and analysis method should have were presented. The results confirmed the fulfilment of the first gap by showing that L-Soft supports all the features with medium emphasis. Method acceptance testing was then performed to confirm the likelihood of the adoption of L-Soft as a teaching method and the results showed that L-Soft has a high likelihood of being adopted as a teaching method by experts, hence fulfilling the second research gap. A formal experiment was then performed in which students were taught the L-Soft method and the results showed that the students can successfully understand and apply the method even within a short period of time, hence fulfilling the third research gap.

Figure 8.6 shows the three validations and the corresponding objectives of the L-Soft method they confirmed.

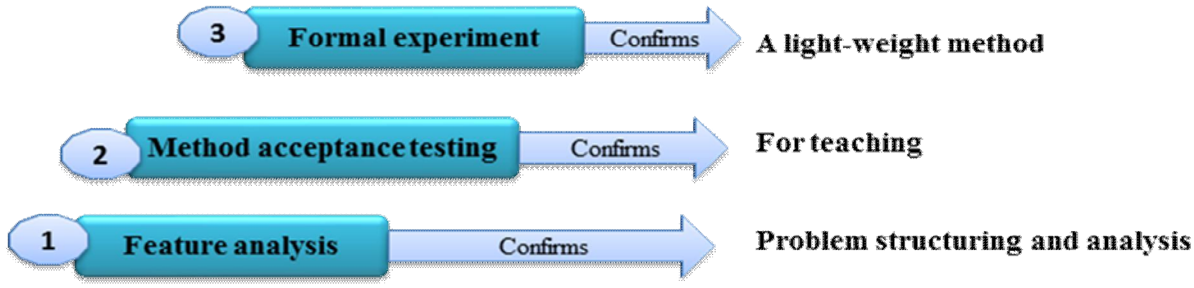


Figure 8.6: The conformance of the three objectives of L-Soft through validations

Hence, the validations satisfied two research objectives presented in Chapter 1 which are objective 2 (i.e. to address the selected problem by formulating a method suitable to be taught to undergraduate software engineering students in the RE course) and objective 4 (i.e. to validate the proposed method). Whereas the first objective was satisfied in Chapters 3 and 4, while the third objective was satisfied in Chapter 7.

In addition, in Chapter 3, a collection of the REE problems was identified and presented in an integrated view. The L-Soft method directly addresses one of the identified REE problems.

8.9 Summary

This chapter reported the empirical validation of the L-Soft method. This validation was based on three different validation methods which are feature analysis, method acceptance testing and formal experiment. The participants of the validation studies were from universities (lecturers and students) as well as from industry (software engineers). The validation process showed affirmative results. Through feature analysis, it was shown that the L-Soft method is appropriate for performing the problem structuring and analysis process. Through method acceptance testing, it was shown that the L-Soft method has a high likelihood of being adopted in practice for the teaching of problem structuring and analysis. Finally through formal experiment, it was shown that the L-Soft method is more efficient and effective, and has a higher likelihood of being adopted in practice for teaching the problem structuring and analysis process rather than an existing analysis method. The validation results are then tied with the research performed in the previous chapters. The next chapter will conclude the research by determining its contributions, limitations and significance.

Chapter 9- Conclusion

9.1 Introduction

This research has examined the importance of REE in software engineering, attracted researchers towards REE problems and emphasized the formulation of light-weight methods for teaching RE in universities. Chapter 1 introduced the research and laid out four distinct research objectives. Chapter 2 introduced and summarized the literature surveys of the state of practice and issues in REE. Chapter 3 analysed REE problems in detail and developed an integrated view of the REE problems. Chapter 4 verified the problems presented in the integrated view through a survey performed on students and lecturers who have studied and taught RE course, and selected the problem of teaching problem structuring and analysis as a research focus. Chapter 5 introduced the area of problem structuring and analysis in RE and presented the research gap that needed to be filled. Chapter 6 presented the L-Soft method proposed to be used for the teaching of problem structuring and analysis in REE. Chapter 7 presented the web-based tool for L-Soft to provide learning support to students using the L-Soft method. Chapter 8 validated the L-Soft method through feature analysis, method acceptance testing and an experimental study.

9.2 Responses to research objectives

- *Research objective 1: To identify and analyse REE problems from the literature and present them in an integrated representation and to select one problem area to be the research focus.*

This objective was first approached in Chapter 2 where REE problems were identified through literature survey. The detailed analysis of the identified problems was performed in which they were arranged into groups by referencing similar issues, classified into categories and then relationships were identified between them.

The output was an integrated view of the REE problems (Figure 3.4) that provides an overview of the relevant information on REE problems. In Chapter 4, the REE problems presented in the integrated view were verified through an investigation performed on students and lecturers who have studied and taught RE course. The investigation results confirmed the problems presented in the integrated view (Section 4.4) and one problem area was selected as the research focus which is the problem of teaching problem structuring and analysis in requirements engineering (Section 4.6).

- *Research objective 2: To address the selected problem by formulating a method suitable to be taught to undergraduate software engineering students in RE course.*

In Chapter 5, a literature survey on the problem of teaching problem structuring and analysis were performed and it was shown that this phase is not explicitly emphasized in RE course but thought to be covered using requirements analysis methods. For students with no industrial experience, it is difficult to perform problem structuring and analysis using these methods. Therefore in Chapter 6, a light-weight method was proposed to be taught to undergraduate students in RE course to enable them to understand and perform the problem structuring and analysis process. The method was called L-Soft as the method has transformed the idea of Soft System methodology into a light-weight (simple structure and easy to apply procedure) RE method. In order to facilitate the understanding and application of the L-Soft method, students were provided with a step-by-step guide on performing the problem structuring and analysis process (Section 6.3.2 and 6.3.3), a glossary of terms used in the method (Section 6.2.1) and solved case studies.

- *Research objective 3: To design and develop a software prototype as a learning support tool to support the proposed method.*

In Chapter 7, the L-Soft method was implemented in a software prototype that acts as a learning support tool. The tool is a web based system to help students in learning and applying the L-Soft method and lecturers in teaching the method. The L-Soft tool aimed at providing two main objectives, first to automate the steps of the method and second, to provide learning support in order to facilitate users to learn, understand and apply the method and to store and retrieve a project's data. The tool is meant to be used by students (tool users) and lecturers (tool admin).

- *Research objective 4: To validate the proposed method.*

In Chapter 8, the L-Soft method was validated. The appropriateness of the method, that is how well it fits the needs of performing the problem structuring and analysis process, was validated by feature analysis performed by software engineers and RE lecturers (Section 8.4.5.1). The statistical analysis results showed that according to the respondents, L-Soft is appropriate for performing the problem structuring and analysis process (the null hypothesis (Ho: The data is fit) was not rejected as the p-value was found to be greater than 0.05 in the Shapiro-Wilk test results). The likelihood of adoption in practice was validated by the perception-based measures of the method evaluation model, through a post-task survey performed by software engineers and RE lecturers (Section 8.4.5.2). The results showed that the L-Soft method has a high likelihood of being adopted in practice for teaching students the problem structuring and analysis process in RE course in universities (as the significance value (two-tailed) was positive for each construct in the one-sample t-tests results). Finally, a formal experiment based on the method evaluation model was performed on undergraduate software engineering students to measure the successfulness of the L-Soft method in terms of efficiency, effectiveness and perceived adoption in practice (Section 8.5).

The L-Soft method was compared to an existing analysis process, and the output was measured by a two-way ANOVA. The results showed that L-Soft was more efficient and effective than the existing analysis method (in terms of all requirement characteristics since the value for all variables was less than 0.05 in the pairwise comparison test). Through post-task survey, it was shown that L-Soft has a high likelihood of being adopted into practice than the existing analysis method (as all comparisons were found to be significantly positive for all variables in the L-Soft method in the one-sample t-tests results).

9.3 Research contributions

This research supports REE and presented the REE problems as an important research area. The contributions of this research are as follows:

a. The development of an integrated view of the REE problems that made it possible for researchers to have a bird's-eye view of all the REE problems together with their analysis information (Figure 3.4, section 3.7). Of the problems presented in the integrated view, some problems have been reported and investigated by researchers while some remain only as reported problems. The reported problems can be seen as hot research areas for RE researchers. Therefore, the main contribution of the integrated view is in its impact on both researchers as well as academics. It attracts researchers by presenting the hot topics for future research in REE and motivates academics to pay more attention and improve the RE course topics and teaching strategies.

b. The development of a light-weight method named L-Soft, based on the idea of soft system methodology, to be taught to undergraduate software engineering students in RE course (Chapter 6, section 6.3). The method covers the two stages of problem structuring and analysis in RE which are, identifying and understanding the problem and its causes, and decomposing, organizing and refining the problem into a set of requirements for a desired system. An important contribution of the L-Soft method is to enable students to relate problem understanding and structuring to requirements extraction. The method is characterized as light weight because it follows one simple structure, it is easy to understand and to apply the procedure, and it uses the terms which students are already familiar with. Probably most significantly, the research demonstrates to the REE community that light-weight methods can be effective tools for training and educating students about RE. Successfully understanding and applying these methods is an important precondition of a successful RE process that can help to provide high quality requirements that adds value to the entire software development lifecycle.

c. The web-based software tool for L-Soft (Chapter 7). The tool is developed to provide learning support to students in many ways such as the introduction and method's highlights are provided to enable students to understand the L-Soft method's underlying concepts, the glossary of terms is provided to make them familiar with the terms used in the method, the "getting started" demo is provided to familiarize students with the working of the tool while the guidelines are provided with each step of the method to facilitate them with the procedures of applying the L-soft method, and the solved case studies are provided to make students understand the procedure of the method.

The tool automates the application of the steps of the method. It also makes it easy for lecturers to teach the method. Although such a tool has a very simple objective of providing automated support to the steps of the method, it has a more fundamental effect in simplifying the method and making the activities more consistent.

9.4 Research limitations

There are several limitations in this research. The L-Soft method is aimed at being a light-weight (easy to understand and apply) RE method for performing the problem structuring and analysis process. However due to the complex nature of RE, students still need to perform a lengthy process and have to master many terms and concepts in order to cover the complete problem structuring and analysis process. To overcome this, attempts were made to simplify the process by using the L-Soft tool, guidelines, templates and solved case studies. In addition, the scope of the L-Soft method was limited to the problem structuring and analysis process only. Students need to be taught and use requirements elicitation activities along with the L-Soft method in order to successfully apply the method and generate requirements. Also it does not cover complete RE process and needs to be accompanied with other RE processes and techniques.

The main concept of the method is based on two viewpoints that are: What it is and what it will be. The idea is to change the workflow of the process so that the problem can be addressed. But in the real projects, it is the fact that customers are normally not happy with the change of way the processes are performed (existing work processes).

However the reason behind using this approach is that we are giving the chance to students (future requirements engineers) to creatively think about the alternate processes that can address the customer's problems, also the changes suggested by them cannot be implemented without approval of customers, and if customer is not happy with the suggestions some other alternatives can be suggested to address the process. Therefore it is assumed that the proposed approach can work efficiently.

L-Soft method and tool have been developed for teaching purposes only to make students understand and perform the problem-based projects (Problem-based projects start when a problem arise that demands a response). The blocking factors in their usage in production are they do not cover the complete RE process, plus they may not applicable to all domains. They only supports problem-based projects but not applicable to other types of projects such as contract-based or game development projects. This can also be seen as a limitation of L-Soft method.

The validations performed to evaluate the L-Soft method are not without limitations. Feature analysis was used as one of the evaluation methods. The features were defined by the researcher, who is also the author of this dissertation. The experimental sessions were also handled by the author who acted as a facilitator. This may suggest a bias in the study outcome. To protect the integrity of the outcome, the author's participation was controlled. All studies managed by a facilitator were managed consistently (i.e. the same level of help or non-help was provided to all participants). Additionally, the results were cross-validated.

In addition, the time period of the formal experiment was fairly short. The example problems used in the study were easy enough to make students understand and solve the problems within a short period of time. The limited follow-up meant that no assessment was done to ascertain whether the knowledge of the problem structuring and analysis process was sustained in the long term. Finally, the characteristics of the generated requirements are subjective; therefore these were first rated by experts and then were measured using specific tests.

9.5 Future work

The reactions to the L-Soft method have been very positive and supportive. Future work in this area includes a closer examination of the deficiencies of the method and tool to improve its practical application. There are several suggestions that can be taken to further this research for future enhancement. The evaluation results (particularly the expert's suggestions and the reported deficiencies) combined with the limitations of this research signify several directions that could be explored. The improvement needs as well as enhancements can be done in the following areas:

L-Soft method and tool:

- The input of the method can be more formal probably in terms of the viewpoints of different stakeholders.
- The current system (existing work processes) should be focused more. There should be a more systematic way to help requirements engineers to refine the work processes.
- The guidelines should be more comprehensive to increase understandability.
- In the requirements presentation step, more requirements properties can be added.
- There can be a simulation-based help for users, in addition to the demo, in the L-Soft tool.

The scope of L-Soft:

- The scope of L-Soft can be extended to cover other types of RE activities. It would be an interesting research problem to map and track the effectiveness of the light-weight method as a training tool with other RE activities. Additional features also could be included to the method to further increase student's understanding.
- Some important requirements engineering elements such as requirements categorization, change management, and requirements classification scheme can be added to allow the method to be used not only for teaching in universities but also in industry settings.
- The L-Soft tool can be enhanced to help users generate requirements specifications for projects.

The evaluation of L-Soft:

- In the formal experiment, each individual was not asked about any problems they faced while using the method. Also, a detailed analysis of each individual's performance was not conducted. Therefore, some information on the students' perceptions of the L-Soft method and tool that might be useful for future research could have been overlooked.
- Since L-Soft was experimented in only one country (Malaysia), there is room to further explore the applicability in other cultural settings.
- The L-Soft method and tool need to be tested by teaching RE course in real class rooms in which the duration should be longer for students to fully understand and apply the method, and more complex case studies used to test the applicability of the method by the students.

This research can be considered as an important step for attracting the attention of researchers towards REE problems and for using light-weight methods in teaching the RE course. Light-weight methods such as this one can obviously play an important role in wider educational and training programs and should be supported with additional types of approaches that have complementary teaching and learning objectives. The examples of such approaches are Lamsveerde's KAOS approach (Darimont, Delor, Massonet, & van Lamsweerde, 1997), problem frames (Jackson, 2005) and i* framework (Franch, 2012).

9.6 Summary

This chapter summarized the overall research. It first described the research results in terms of the fulfilment of the research objectives. The main research contributions are presented and the limitations of the research are described. The future work is then proposed to enhance and improve the research.

References

Abran, A., Moore, J. W., Bourque, P., Dupuis, R., & Tripp, L. L. (2004). *Guide to the software engineering body of knowledge: 2004 version*. IEEE Computer Society, Los Alamitos, CA; Tokyo.

Ackoff, R.L. (1979). The future of operational research is past. *Journal of Operational Research Society, 30, 189-199.*

Adroin, W. R. (2000). *Developing and deploying software engineering courseware in an adaptable curriculum framework.* Paper presented at the 22nd international conference on Software engineering, New York, NY, USA.

Agre, G.P. (1982). The concept of problem. *Educational Studies, 13*(2), 121-142.

Al-Ani, B., & Yusop, N. (2004). *Role-playing, group work and other ambitious teaching methods in a large requirements engineering course.* Paper presented at the 1th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems (ECBS '04)

Alexander, I.F., & Stevens, R. (2002). *Writing better requirements* (1 ed.). Addison-Wesley.

Andersen, B., & Fagerhaug, T. (2006). *Root cause analysis: simplified tools and techniques* (2nd ed.). ASQ Quality Press.

Anton, A.I. (1996). *Goal-based requirements analysis.* Paper presented at the Second IEEE International Conference on Requirements Engineering (ICRE '96), Colorado, USA.

Armarego, J., & Minor, O. (2005). *Studio Learning of Requirements: towards aligning teaching to practitioner needs.* Paper presented at the International workshop on Requirements Engineering Education & Training, Paris.

Avison, DE, Golder, PA, & Shah, HU. (1992). Towards an SSM toolkit: Rich picture diagramming. *European Journal of Information Systems, 1*(6), 397-408.

Badal, M.A. (2006). *Strategic management: problem structuring methods.* Technical University of Denmark.

Barnes, R. J., Gause, D. C., & Way, E. C. (2008). *Teaching the Unknown and the Unknowable in Requirements Engineering Education.* Paper presented at the Requirements Engineering Education and Training.

Barroca, L., Fiadeiro, J. L., Jackson, M., Laney, R., & Nuseibeh, B. (2004). Problem frames: A case for coordination. *Coordination Models and Languages*, 5-19.

Batra, A. (2010). Compound Metrices in Web Analytics. Retrieved from http://webanalysis.blogspot.com/#axzz2qBWsr5Nn

Beatty, J., & Agouridas, V. (2007). *Developing Requirements Engineering Skills: A Case Study in Training Practitioners*. Paper presented at the International Workshop on Requirements Engineering and Training (REET2007) India Habitat Centre, New Delhi.

Berenbach, B. (2005). *A hole in the curriculum*. Paper presented at the International workshop on Requirements Engineering Education & Training, Paris.

Biasutti, M. (2011). The student experience of a collaborative e-learning university module. *Computers & Education, 57*(3), 1865-1875.

Bray, I. K. (2002). *An Introduction to Requirements Engineering*. Addison Wesley.

Brooks, F.P. (1987). No silver bullet: Essence and accidents of software engineering. *IEEE computer, 20*(4), 10-19.

Caldiera, V. R. B. G., & Rombach, H. D. (1994). The goal question metric approach. *Encyclopedia of software engineering, 2*(1994), 528-532.

Callele, D., & Makaroff, D. (2006). Teaching requirements engineering to an unsuspecting audience. *ACM SIGCSE Bulletin, 38*(1), 433-437.

Charmaz, K., & Henwood, K. (2008). Grounded theory. *The SAGE handbook of qualitative research in psychology* (pp. 240–259). Newbury Park: Sage Publications.

Checkland, P., & Scholes, J. . (1999). *Soft systems methodology in action*. Chichester, West Sussex: Wiley.

Connor, A. M., Buchan, J., & Petrova, K. (2009). *Bridging the Research-Practice Gap in Requirements Engineering through Effective Teaching and Peer Learning*. Paper presented at the Sixth International Conference on Information Technology: New Generations.

Crnkovic, Gordana Dodig. (2010). Constructive research and info-computational knowledge generation. *Model-Based Reasoning in Science and Technology* (pp. 359-380): Springer.

Daniels, M., Carbone, A., Hauer, A., & Moore, D. (2007). S*tructured problem solving in engineering education*. Paper presented at the 37th ASEE/IEEE Frontiers in Education Conference, Milwaukee, WI.

Darimont, R., Delor, E., Massonet, P., & Van L. A. (1997). *GRAIL/KAOS: An environment for goal-driven requirements engineering*. Paper presented at the 19th international conference on Software engineering.

Davis, A.M., Hickey, A.M., & Chamillard, A. T. (2005). *Moving Beyond the Classroom: Integrating Requirements Engineering Research & Education to Improve Practice*. Paper presented at the Workshop on Requirements Engineering Education & Training, Paris.

Differding, C., Hoisl, B., & Lott, C. M. (1996). *Technology package for the goal question metric paradigm*: Fachbereich Informatik, Univ.

Easton, V. J., & McColl, J. H. Statistics Glossary v1.1. Retrieved 20th March, 2013, from http://www.stats.gla.ac.uk/steps/glossary/hypothesis_testing.html#1sampt

Eberlein, A. P. G. (1997). *Requirements Acquisition and Specification for Telecommunication Services.* (PhD Thesis), University of Wales Swansea.

Eden, C. (2004). Analyzing cognitive maps to help structure issues or problems. *European Journal of Operational Research, 159*(3), 673-686.

El-Shamy, S. (2001). *Training games: Everything you need to know about using games to reinforce learning*. Sterling, VA: Stylus Publ.

El Emam, K., & Madhavji, N.H. (1995). *A field study of requirements engineering practices in information systems development.* Paper presented at the Second IEEE International Symposium on Requirements Engineering, York, England.

Ellspermann, S.J., Evans, G.W., & Basadur, M. (2007). The impact of training on the formulation of ill-structured problems. *Omega, 35*(2), 221-236.

España, S., Condori-Fernandez, N., González, A., & Pastor, Ó. (2010). An empirical comparative evaluation of requirements engineering methods. *Journal of the Brazilian Computer Society*, *16*(1), 3-19.

Evans, J. R. (1991). *Creative thinking in the decision and management sciences*: Cincinnati, OH: South-Western Publishing.

Ferrari, R., Miller, J. A., & Madhavji, N. H. (2010). A controlled experiment to assess the impact of system architectures on new system requirements. *Requirements Engineering, 15*(2), 215-233.

Firesmith, D. (2004). Prioritizing requirements. *Journal of Object Technology, 3*(8), 35-47.

Franch, X. (2012). *The i∗ framework: The way ahead.* Paper presented at the Sixth International Conference on Research Challenges in Information Science (RCIS), 2012.

Gause, D.C., & Weinberg, G.M. (1990). *Are your lights on? :How To Figure Out What The Problem Really is.* Weinberg & Weinbrg.

Gibson, J.P. (2000). *Formal requirements engineering: Learning from the students.* Paper presented at the Software Engineering Conference, Australia.

Gonzales, R.M., & White, S. (1999). *The Effectiveness of Teaching System Requirements Analysis*. Paper presented at the NMSU Engineering Education Conference, New Mexico.

Gorschek, T., & Wohlin, C. (2006). Requirements abstraction model. *Requirements Engineering, 11*(1), 79-101.

Grimán, A., Pérez, M., Mendoza, L., & Losavio, F. (2006). Feature analysis for architectural evaluation methods. *Journal of Systems and Software, 79*(6), 871-888.

Hoffmann, A. (2008). *Teaching Soft Facts in Requirements Engineering Using Improvisation Theatre Techniques*. Paper presented at the Third international workshop on Multimedia and Enjoyable Requirements Engineering - Beyond Mere Descriptions and with More Fun and Games, Barcelona, Catalunya.

Huijs, C., Sikkel, K., & Wieringa, R. (2005). *Mission 2 Solution: Requirements Engineering Education as Central Theme in the BIT Programme*. Paper presented at the Requirements Engineering Education & Training, Paris.

Jackson, M. (1999). Problem analysis and structure. *Engineering Theories of Software Construction*, 3-20.

Jackson, M. (2005). Problem frames and software engineering. *Information and software technology, 47*(14), 903-912.

Jacobson, I. (1992). *Object-oriented software engineering: a use case driven approach*: Pearson Education, India.

Jiang, L. (2005). *A framework for the requirements engineering process development.* (Ph.D.), University of Calgary (Canada), Canada. Retrieved from http://proquest.umi.com/pqdweb?did=1003855461&Fmt=7&clientId=18803&RQT=309&VName=PQD

Jiang, L., Eberlein, A., & Far, B.H. (2005). *Combining Requirements Engineering Techniques–Theory and Case Study*. Paper presented at the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05).

Johnson, B., & Christensen, L. (2003). Educational research: Quantitative, qualitative, and mixed approaches. Retrieved 1st jan 2014, from www.southalabama.edu/coe/bset/johnson/dr_johnson/2lectures.htm

Kitchenham, B. (1996). DESMET: A method for evaluating Software Engineering methods and tools (Vol. Technical Report TR96-09). Keele Staffordshire: Department of Computer Science, University of Keele.

Kline, P. (2000). *Handbook of Psychological Testing* (2nd ed.). London: Routledge.

Kotonya, G., & Sommerville, I. (1998). *Requirements Engineering Processes and Techniques*: John Wiley & Sons.

Kulak, D., & Guiney, E. (2004). *Use cases: requirements in context* (Second ed.). Addison-Wesley Professional.

LeBlanc, R. J., Sobel, A., Diaz-Herrera, J. L., & Hilburn, T. B. (2006). Software engineering 2004: curriculum guidelines for undergraduate degree programs in software engineering. *ACM/IEEE-CS Joint Task Force on Computing Curricula*: IEEE Computer Society.

Leffingwell, D., & Widrig, D. (1999). *Managing software requirements: a unified approach*. Addison-Wesley Professional.

Leite, J. C. S. P. (1989). *Viewpoint analysis: a case study*. Paper presented at the 5th international workshop on Software specification and design (IWSSD '89), New York, NY, USA.

Li, Z. (2008). *Progressing problems from requirements to specifications in problem frames.* Paper presented at the IWAAPF'08.

McPhee, C., & Eberlein, A. (2000). *An Approach to Evaluating Requirements Engineering Methods for Applicability to Time-to-Market Projects.* Paper presented at the 13th International Conference on Software & Systems Engineering and their Applications (ICCSEA2000), Paris, France.

Melonfire, Contributor. (2007). Five common errors in requirements analysis (and how to avoid them). Retrieved 25th nov, 2011, from http://www.techrepublic.com/article/five-common-errors-in-requirements-analysis-and-how-to-avoid-them/6146544

Memon, R.N., Ahmad, R., & Salim, S.S. (2013). Requirements Engineering Education (REE): Problems and Recommendations for RE course implementations. *Malaysian Journal of Computer Science*, *26*(4), 294-311.

Memon, R.N., Salim, S.S., & Ahmad, R. (2012). *Identifying Research Gaps in Requirements Engineering Education: An Analysis of a Conceptual Model and Survey Results*. Paper presented at the IEEE Conference on Open Systems 2012, Kuala Lumpur, Malaysia.

Memon, R.N., Salim, S.S., & Ahmad, R. (2013). Analysis and classification of problems associated with requirements engineering education: Towards an integrated view. *Arabian Journal for science and Engineering*, *39*(3), 1923-1935: Springer.

Mingers, J., & Rosenhead, J. (2004). Problem structuring methods in action. *European Journal of Operational Research*, *152*(3), 530-554.

Minor, O., & Armarego, J. (2005). Requirements Engineering: a Close Look at Industry Needs and Model Curricula. *Australasian Journal of Information Systems, 13*(1), 192.

Moody, D. L., Sindre, G., Brasethvik, T., & Sølvberg, A. (2003). *Evaluating the quality of information models: empirical testing of a conceptual model quality framework*. Paper presented at the 25th International Conference on Software Engineering (ICSE '03), Washington, DC, USA.

Moody, D.L. (2002). *Complexity effects on end user understanding of data models: An experimental comparison of large data model representation methods*. Paper presented at the 10th European Conference on Information Systems (ECIS), Gdansk, Poland.

Moody, D.L. (2003). *The method evaluation model: a theoretical model for validating information systems design methods*. Paper presented at the 11th European Conference on Information Systems (ECIS) Naples, Italy.

Nguyen, L., Armarego, J., & Swatman, P. (2002). Understanding Requirements Engineering: a Challenge for Practice and Education. *School Working Papers Series 2002*.

Niazi, M.K. (2002). Improving the Requirements Engineering Process through the Application of a Key Process Areas Approach. *AWRE02*.

Pfleeger, S.L. (1995). Experimental design and analysis in software engineering. *Annals of Software Engineering, 1*(1), 219-253.

Pidd, M. (1988). From problem-structuring to implementation. *Journal of the Operational Research Society, 39*(2), 115-121.

Pidd, M., & Woolley, R.N. (1980). Four views on problem structuring. *Interfaces, 10*(1), 51-54.

Regev, G., Gause, D. C., & Wegmann, A. (2009). Experiential learning approach for requirements engineering education. *Requirements Engineering, 14*(4), 269-287.

Rittel, H.W.J., & Webber, M.M. (1973). Dilemmas in a general theory of planning. *Policy sciences, 4*(2), 155-169.

Robertson, S., & Robertson, J. (2012). *Mastering the Requirements Process- Getting Requirements Right* (3 ed.). Addison-Wesley Professional.

Rogers, D., Stratton M.J., and R. E. King. (1999). Manufacturing Education Plan: 1999 Critical Competency Gaps. *Dearborn, Mich.: Society of Manufacturing Engineers*.

Rolland, C., Souveyet, C., & Achour, C. B. (1998). *Guiding goal modeling using scenarios*. Paper presented at the IEEE Transactions on Software Engineering.

Rosca, D. (2000). *An active/collaborative approach in teaching requirementsengineering*. Paper presented at the 30th Annual Frontiers in Education.

Rosenhead, J. (1996). What's the problem? An introduction to problem structuring methods. *Interfaces, 26*(6), 117-131.

Ross, D.T. (1977). Structured analysis (SA): A language for communicating ideas. *IEEE Transactions on Software Engineering*(1), 16-34.

Ross, D.T., & Schoman Jr, K.E. (1977). Structured analysis for requirements definition. *IEEE Transactions on Software Engineering*(1), 6-15.

Sawyer, P., Sommerville, I., & Viller, S. (1997). Requirements process improvement through the phased introduction of good practice. *Software Process Improvement and Practice, 3*(1), 19-34.

Scheinholtz, L. A. (2007). *What Are Employers Really Looking For?* Paper presented at the 2nd International Workshop on Requirements Engineering Education and Training, India Habitat Center, New Delhi.

Schön, D.A. (1987). *Educating the reflective practitioner: Toward a new design for teaching and learning in the professions*: Jossey-Bass San Francisco.

Schwenk, C., & Thomas, H. (1983). Formulating the mess: The role of decision aids in problem formulation. *Omega, 11*(3), 239-252.

Seddon, F., & Biasutti, M. (2009). Evaluating a music e-learning resource: The participants' perspective'. *Computers & Education, 53*(3), 541-549.

SEI. (1991). Software Engineering Institute Requirements Engineering Project *Requirements Engineering and Analysis Workshop Proceedings* (Vol. Technical Report CMU/SEI-91-TR-30, Software Engineering Institute, Carnegie Mellon University).

Sidky, A.S., Sud, R.R., Bhatia, S., & Arthur, J.D. (2002). Problem Identification and Decomposition within the Requirements Generation Process *Technical report TR-02-31*: Department of Computer Science, Virginia Polytechnic Institute & State University.

Smith, R. (2009). *Gameplay to introduce and reinforce requirements engineering good practices.* (D.P.S.), Pace University, United States -- New York. Retrieved from http://proquest.umi.com/pqdweb?did=1853239111&Fmt=7&clientId=18803&RQT=309&VName=PQD

Smith, R., & Gotel, O. (2007). *RE-O-POLY: A Game to Introduce Lightweight Requirements Engineering Good Practices.* Paper presented at the International Workshop on Requirements Engineering and Training, India Habitat Center, New Delhi.

Sommerville, I., & Kotonya, G. (1998). *Requirements engineering: processes and techniques*. New York, NY, USA: John Wiley & Sons, Inc.

Sorensen, L., & Valqui Vidal, R.V. (2008). Evaluating six soft approaches. *EAWP: Documentos de trabajo en análisis económico= Economical Analysis Working Papers, 7*(9), 1-0.

Strauss, A. L. , & Corbin, J. M. . (1998). *Basics of qualitative research: Techniques and procedures for developing grounded theory*: Sage Publications, Inc.

Svahnberg, M., & Gorschek, T. (2005). *Multi-perspective Requirements Engineering Education with focus on Industry Relevance*. Paper presented at the Workshop on Requirements Engineering Education & Training, Paris.

Thayer, R.H., Bailin, S.C., & Dorfman, M. (1997). *Software Requirements Engineerings*: IEEE Computer Society Press.

UTS. (2010). Hand Book: Advances in Requirements Engineering. May 2011, from http://www.handbook.uts.edu.au/subjects/32550.html

Van L. A. (2001). *Goal-oriented requirements engineering: A guided tour*. Paper presented at the Fifth IEEE International Symposium on Requirements Engineering, Toronto, Ont.

Vidal, R.V.V. (2002). *Dealing with problematic situations*: IMM working paper, DTU, Lyngby.

Volkema, R.J., & Evans, J.R. (1995). Creativity in MS/OR: managing the process of formulating the problem. *Interfaces, 25*(3), 81-87.

Wahono, R. S. (2003). *Analyzing requirements engineering problems*. Paper presented at the IECI Japan Workshop 2003, Chofu Bunka Kaikan Tazukuri, Japan.

Wallace, C., Wang, X., & Bluth, V. (2006). *Instruction in Problem Structuring and Analysis Through Problem Frames*. Paper presented at the Proceedings of the 19th Conference on Software Engineering Education & Training (CSEET'06).

Wang, W., Hufnagel, S., Hsia, P., & Yang, S.M. (1992). *Scenario driven requirements analysis method*. Paper presented at the Second International Conference on Systems Integration (ICSI '92), Morristown, NJ.

Wieringa, R., & Ebert, C. (2004). RE'03: Practical Requirements Engineering Solutions. *IEEE Software*, 16-18.

Wohlin, C., & Regnell, B. (1999). *Achieving industrial relevance in software engineering education*. Paper presented at the 12th Conference on Software Engineering Education and Training, New Orleans, LA.

Woolley, RN, & Pidd, M. (1981). Problem structuring:a literature review. *Journal of the Operational Research Society, 32*(3), 197-206.

Yeh, R.T., & Zave, P. (1980). Specifying software requirements. *Proceedings of the IEEE, 68*(9), 1077-1085.

Yusop, N., Mehboob, Z., & Zowghi, D. (2007). *The Role of Conducting Stakeholder Meetings in Requirements Engineering Training*. Paper presented at the Requirements Engineering Education & Training, India.

Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys (CSUR), 29*(4), 315-321.

Zowghi, D. (2009). *Teaching Requirements Engineering to the Baháí Students in Iran who are Denied of Higher Education.* Paper presented at the Fourth International Workshop on Requirements Engineering Education and Training (REET).

Appendix A - Students' questionnaire

I am a PhD student in the Faculty of Computer Science & Information Technology. This study is a part of research carried out in partial fulfilment of requirements for the degree of Doctor of Philosophy (PhD) from the University of Malaya (UM), Malaysia. This study aims to find out students' perceptions of the Requirements Engineering (RE) course and to identify issues and problems related to RE courses taught in universities. All of the information acquired from this study will be used solely for academic purposes. Your participation and co-operation in this study are greatly appreciated.

Thank you.

Rafia Naz Memon

WHA090011

I am a first/second/third year student studying in UM

Scale: VS = Very Sufficient
S = Sufficient
NVS = Not Very Sufficient
I = Insufficient

Part One

Please tick √ or fill in the blanks where applicable.

1) Answer the following questions based on whether these Requirements Engineering elements have been taught to you in class. If they have, please rate the degree to which the Requirements Engineering elements have been taught to you to sufficiently prepare you for performing them in real projects.

a.	Basic Requirements Engineering concepts (e.g. requirements elicitation, identifying stakeholders, requirements verification, inspections, requirements documents etc)	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
b.	Requirements elicitation techniques (e.g. interviews, questionnaire, focus groups, surveys, observations, prototypes, etc.)	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
c.	Requirements analysis techniques (e.g. Structured analysis, view point-oriented analysis, object-oriented analysis etc.)	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
d.	Requirements modelling techniques (e.g. structural modelling, behavioural (functional) modelling etc.)	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
e.	Techniques to involve customers in each phase of the project	○ **YES**		**NO** ○
		○VS	○ S	
		○NVS	○ I	
f.	Techniques to deal with incomplete requirements provided by customers when starting a new project	○ **YES**		**NO** ○
		○VS	○ S	
		○NVS	○ I	
g.	Techniques to communicate with stakeholders	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
h.	Creating requirements specification documents	○ **YES**		**NO** ○

		○ VS	○ S	
		○NVS	○ I	
i.	Understanding, analysing and structuring initially-presented customer requirements (problem structuring)	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
j.	Techniques to deal with customers' changing requirements during the project	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
k.	Techniques to deal with customers' unrealistic expectations of the project	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
l.	Others, please specify			

2) Have you been provided with experience of performing Requirements Engineering activities for industrial projects?	○ **YES**		**NO** ○
	○ VS	○ S	
	○NVS	○ I	
3) Have you been given class projects to practice Requirements Engineering activities?	○ **YES**		**NO** ○
	○ VS	○ S	
	○NVS	○ I	

4) Have you been taught to deal with the following Requirements Engineering challenges?

a.	Resolving conflicts during a project	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	
b.	Defining the scope of a project	○ **YES**		**NO** ○
		○ VS	○ S	
		○NVS	○ I	

c.	Admitting mistakes (if done in understanding requirements or in creating requirements specifications)	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
d.	Defining expected system behaviour in a combination of user, systems and data states	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
e.	Producing outputs suitable for a diverse audience	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
f.	Giving attention only to functional requirements	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
g.	Increased short-term cost	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
h.	Lack of awareness, maturity and guidance	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
i.	Understanding environment behaviour, and identifying and clarifying requirements in a volatile and uncertain environment	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
j.	Ambiguity, uncertainty, confusion, fear, time pressure, insufficient rigour, collaboration, and corporate politics	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
k.	Facilitating decisions to stakeholders	○ **YES**		**NO** ○
		○ VS	○ S	
		○ NVS	○ I	
l.	If others. Please specify			

5) Which approaches has your lecturer used to teach the Requirements Engineering course? (You can tick more than one answer)

a) Lectures

b) Labs

c) Presentations

d) Group discussions

e) Other, please specify____

Part Two (a)

In your opinion, which of the following Requirements Engineering issues are difficult to understand? Tick the ones you found difficult.

Understanding Requirements Engineering concepts and techniques	
Working on Requirements Engineering tools	
Communicating with customers	
Performing requirements elicitation	
Understanding, analysing and structuring initially-presented customer requirements (problem structuring)	
Dealing with customers' changing requirements	
Dealing with incomplete requirements provided by customers when starting a new project	
Dealing with customers' unrealistic expectations of the project	
Involving customers in each phase of the project	
Producing good quality requirements specifications	
Dealing with other complex Requirements Engineering challenges (those reported in question 4 of Part One)	

Part Two (b)

1) How have you found the Requirements Engineering course?

O Very interesting	O Interesting	OAverage	O Boring	O Very Boring

2) Do you think that Requirements Engineering is an important subject to be taught in universities?

O Very important	O Important	O Fairly important	O Not very	O Not at all

3) Did you have any industrial experience of performing Requirements Engineering activities (e.g. during internships, as part of the final-year project requirements, in research)?

a) No b) Yes c) Don't know

If yes, do you think that the universities are teaching students Requirements Engineering courses according to industry requirements?

○ Very often	○ Often	○ Sometimes	○ Seldom	○ Not at all

4) Will you choose Requirements Engineering as a profession in the future?

a) Yes b) No c) Don't know

5) Do you feel that you have been taught the Requirements Engineering course in sufficient depth in class?

○ Extremely Sufficient	○ Very Sufficient	○ Sufficient	○ Not very sufficient	○ Insufficient

Part Two (c)

1) Have you worked with Requirements Engineering tools during the Requirements Engineering course?

a) Yes b) No c) Don't know

If yes, write down the names of Requirements Engineering tools that you have used.

____

____

____

____

____

2) Write down the problems you faced during the Requirements Engineering course.

____

____

____

____

____

____

3) What are your suggestions for improving the Requirements Engineering course?

____

____

____

____

____

____

Thank you for your time

Appendix B - Lecturers' questionnaire- I

Faculty of Computer Science & Information Technology

University of Malaya

I am a PhD student in the faculty of Computer Science and Information Technology, University of Malaya.

This study aims to find out lecturers' perceptions of the Requirements Engineering (RE) course and to identify issues and problems related to RE courses taught in universities. All of the information acquired from this study will be used solely for academic purposes. Your participation and co-operation in this study are greatly appreciated.

Thank you.

Rafia Naz Memon
WHA090011

I am Lecturer/Senior Lecturer/Associate Professor/Professor at ……………..

Part One

1) How long have you taught the Requirements Engineering (RE) course?
   a) One Semester
   b) Two Semesters
   c) Three Semesters
   d) Other, please specify ________________________________

2) Please tick √ or fill in the blanks where applicable.

Scale: HE= Highly Emphasized FE=Fairly Emphasized LE=Less Emphasized NE=Not Emphasized

i.	Please rate the degree to which the following RE elements are emphasised in the RE syllabus you have taught. If an element was given less emphasis or was not emphasised at all, please provide a reason (e.g., this element is not as important as the others, it was not included in the syllabus provided by the university administration, it was difficult to teach using the resources provided by the university.).				
a.	Basic RE concepts (e.g., requirements elicitation, identification of stakeholders, requirements verification, inspections, requirements documents.)	HE	FE	LE	NE
	Please elaborate on your response:				
b.	Requirements elicitation techniques (e.g., interviews, questionnaires, focus groups, surveys, observations, prototypes, etc.)	HE	FE	LE	NE
	Please elaborate on your response:				
c.	Requirements analysis techniques (e.g., structured analysis, view point-oriented analysis, object-oriented analysis, etc.)	HE	FE	LE	NE
	Please elaborate on your response:				

d.	Requirements modelling techniques (e.g., structural modelling, behavioural (functional) modelling, etc.)	HE	FE	LE	NE
	Please elaborate on your response:				
e.	Techniques used to involve customers in each phase of the project	HE	FE	LE	NE
	Please elaborate on your response:				
f.	Techniques used to deal with incomplete requirements provided by customers when starting a new project	HE	FE	LE	NE
	Please elaborate on your response:				
g.	Techniques used to communicate with stakeholders	HE	FE	LE	NE
	Please elaborate on your response:				
h.	Creating requirements specification documents	HE	FE	LE	NE
	Please elaborate on your response:				
i.	Understanding, analysing and structuring the customer's initial requirements (problem structuring)	HE	FE	LE	NE
	Please elaborate on your response:				
j.	Techniques used to deal with customers' changing requirements during the project	HE	FE	LE	NE
	Please elaborate on your response:				
k.	Techniques used to deal with customers' unrealistic expectations of the project	HE	FE	LE	NE
	Please elaborate on your response:				

l.	Other (please specify)				
ii.	Have you provided students with the experience of performing RE activities for industrial projects?	HE	FE	LE	NE
Please elaborate on your response:					
iii.	Have you set class projects in order to enable students to practise RE activities?	HE	FE	LE	NE
Please elaborate on your response:					
iv.	Have you taught students how to deal with the following RE challenges?				
a.	Resolving conflicts during a project	HE	FE	LE	NE
	Please elaborate on your response:				
b.	Defining the scope of a project	HE	FE	LE	NE
	Please elaborate on your response:				
c.	Admitting mistakes (if made in understanding requirements or in creating requirements specifications)	HE	FE	LE	NE
	Please elaborate on your response:				
d.	Defining expected system behaviour in a combination of user, system and data states	HE	FE	LE	NE
	Please elaborate on your response:				

e.	Producing outputs which are suitable for a diverse audience	HE	FE	LE	NE
	Please elaborate on your response:				
f.	Paying attention to functional requirements only	HE	FE	LE	NE
	Please elaborate on your response:				
g.	Increased short-term cost	HE	FE	LE	NE
	Please elaborate on your response:				
h.	Lack of awareness, maturity and guidance	HE	FE	LE	NE
	Please elaborate on your response:				
i.	Understanding the behaviour of an environment, and identifying and clarifying requirements in a volatile and uncertain environment	HE	FE	LE	NE
	Please elaborate on your response:				
j.	Ambiguity, uncertainty, confusion, fear, time pressure, insufficient rigour, collaboration and corporate politics	HE	FE	LE	NE
	Please elaborate on your response:				
k.	Facilitating decision-making with stakeholders	HE	FE	LE	NE
	Please elaborate on your response:				
l.	Other (please specify)				

3) Which approaches have you used to teach the RE course? (Select all that apply)

a) Lectures

b) Labs

c) Presentations

d) Group discussions

e) Other, please specify____

Part Two (a)

In your opinion, which of the following RE issues are difficult to teach? Select all that apply.

Understanding RE concepts and techniques	
Please elaborate on your response:	
Working with RE tools	
Please elaborate on your response:	
Communicating with customers	
Please elaborate on your response:	
Performing requirements elicitation	
Please elaborate on your response:	
Understanding, analysing and structuring the customer's initial requirements (problem structuring)	
Please elaborate on your response:	
Dealing with customers' changing requirements	
Please elaborate on your response:	
Dealing with incomplete requirements provided by customers	

when starting a new project	
Please elaborate on your response:	
Dealing with customers' unrealistic expectations of the project	
Please elaborate on your response:	
Involving customers in each phase of the project	
Please elaborate on your response:	
Producing good-quality requirements specifications	
Please elaborate on your response:	
Dealing with other complex RE challenges (e.g., those reported in question iv of Part One)	
Please elaborate on your response:	

Part Two (b)

1) How have you found teaching RE?

○ Extremely easy	○ Very easy	○ Fairly easy	○ Easy	○ Not very easy

2) How important do you think it is that RE is taught in universities?

○ Very important ○ Important ○ Fairly important ○ Not very important ○ Not at all impo

3) How well are you aware of the industry requirements for performing RE?

○ Fully aware	○ Strongly aware	○ Aware	○ A little aware	○ Unaware

4) Do you think that the way that the RE course has been taught to students has motivated them to choose RE as a future profession?

a) Yes b) No c) I don't know

5) Do you think that universities are teaching their RE courses to students in accordance with industry requirements?

a) Yes b) No c) I don't know

6) How sufficient do you think that the infrastructure provided by universities and colleges is with regard to teaching RE?

○ Extremely sufficient	○ Very sufficient	○ Sufficient	○ Not very sufficient	○ Insufficien

Part Two (c)

1) Which RE tools do you use in teaching the RE course?

____

____

____

____

____

2) Write down the problems you have faced while teaching the RE course.

____

____

____

____

____

3) What are your suggestions for improving the RE course?

____

____

____

____

____

Appendix C - Lecturer's questionnaire - II

1) From the RE issues mentioned in the previous section, the issue "Teaching students to understand, analyse and structure the customer's initial requirements (problem structuring)" has been selected as a focus for this research, as it is an important issue that has not yet been investigated in the literature. Please give your comments, feedback and suggestions regarding this selected research focus.

____

____

____

____

____

____

2) Do you believe that this is an area that requires attention in research?

____

____

____

____

____

____

Appendix D - Lecturers' and software engineers' post-task questionnaire

I am a PhD student in the faculty of Computer Science and Information Technology, University of Malaya, Malaysia. This study aims to collect the lecturer's perceptions about L-Soft method. It is advisable that you examine the L-Soft method and tool before answering the questionnaire. All the information acquired from this study will be used solely for academic purposes. Your participation and Co-operation in this study are greatly appreciated.

* Required

Your Name - Please share your name (if you want)

Your institution name and address *- Please write the name and address of institution you are teaching in.

Your current position * - Please share on which position you are working in your institution.

Your working experience * - Please share how long you are working as an academic.

Please rate the following questions according to your perception about L-Soft.

How do you perceive L-Soft method?

I found the procedure for applying the method complex and difficult. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that this method would reduce the effort required to analyse and structure customer's problem into set of requirements. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

The requirements extraction process using this method is difficult for students to understand. *

- Strongly agree

- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I found that the tool based on this method is difficult to be used. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

This method would make it easier for lecturers to verify the extracted requirements. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that the method is easy to be understood. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I found the method to be useful in teaching problem structuring and analysis process. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

By using this method, it is more difficult to analyse and structure the problem. *

- Strongly agree

- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that it is difficult to apply the method for the given problem. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I would definitely not use this method to teach problem structuring and analysis process in requirements engineering. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found the given guidelines are clear and easy to understand. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method does not provide an effective solution to the problem of teaching problem structuring and analysis. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method is an improvement to the standard requirements analysis methods taught in RE course. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I am not confident that this method can be used in teaching RE course. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Use of tool makes it easier to teach to apply the method. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I intend to use this method in preference to the standard requirements engineering methods taught in RE course if I have to teach students to extract the requirements from customer's problem in the future. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

According to your opinion, estimate the degree to which L-Soft supports the following features. *

	High emphasis	Medium emphasis	Low emphasis	No emphasis
Have sound theoretical basis	○	○	○	○
Help beginners	○	○	○	○

	High emphasis	Medium emphasis	Low emphasis	No emphasis
Understanding problem	○	○	○	○
Identifying stakeholders	○	○	○	○
Encourage user participation	○	○	○	○
Decomposing problem	○	○	○	○
Structuring the problem	○	○	○	○
Examining the current system	○	○	○	○
Highlighting the changes needed	○	○	○	○
Use business concerns (goals) to derive requirements	○	○	○	○
Use scenarios to derive requirements	○	○	○	○
Generating requirements from elements of the decomposed problem set	○	○	○	○
Collecting requirements from multiple viewpoints	○	○	○	○
Generating requirements at different abstraction levels	○	○	○	○
Define potential requirements	○	○	○	○
Uniquely identifying each requirement	○	○	○	○
Use checklists for requirements analysis	○	○	○	○

	High emphasis	Medium emphasis	Low emphasis	No emphasis
Prioritizing requirements	○	○	○	○
Define standard requirement structure	○	○	○	○
Define standard templates for requirements description	○	○	○	○

In your opinion, which of the following areas have deficiencies and require improvement in L-Soft.

- ☐ The concepts used for the method
- ☐ The steps of the method
- ☐ The procedure used to apply the method
- ☐ The L-Soft tool
- ☐ The guidelines provided with the method
- ☐ The solved case studies
- ☐ The Getting started demo
- ☐ Others

Please elaborate on your selection.

Please write any other deficiency/problem you found in the method or any suggestion for improving the method.

Submit

Appendix E - Student's post-task questionnaire on L-Soft

I am a PhD student in the faculty of Computer Science and Information Technology, University of Malaya, Malaysia. This study aims to collect the student's perceptions about L-Soft method. It is advisable that you learn and apply the L-Soft method before answering the questionnaire. All the information acquired from this study will be used solely for academic purposes. Your participation and Co-operation in this study are greatly appreciated.

* Required

Which year you are studying in? *

- 1st
- 2nd
- 3rd

Do you have any experience of working on industrial projects? *

- Yes
- No

Please rate the following questions according to your perception about L-Soft.

How do you perceive L-Soft method?

I found the procedure for applying the method complex and difficult. *

- Strongly agree
- Agree

Neither agree nor disagree

Disagree

Strongly disagree

I found that this method would reduce the effort required to analyse and structure customer's problem into set of requirements. *

Strongly agree

Agree

Neither agree nor disagree

Disagree

Strongly disagree

The requirements extraction process using this method is difficult for students to understand. *

Strongly agree

Agree

Neither agree nor disagree

Disagree

Strongly disagree

Overall, I found that the tool based on this method is difficult to be used. *

Strongly agree

Agree

Neither agree nor disagree

Disagree

Strongly disagree

This method would make it easier for students to verify the extracted requirements. *

Strongly agree

Agree

Neither agree nor disagree

Disagree

Strongly disagree

I found that the method is easy to be understood. *

Strongly agree

- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I found the method to be useful in learning problem structuring and analysis process. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

By using this method, it is more difficult to analyse and structure the customers' problem. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that it is difficult to apply the method for the given problem. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I would definitely not use this method to perform problem structuring and analysis process in requirements engineering. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that the given guidelines are clear and easy to understand. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method does not provide an effective solution to the problem of teaching problem structuring and analysis. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method is an improvement to the standard requirements analysis methods taught in RE course. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I am not confident that I am now competent to apply this method in practice. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

By using the software tool it is easier to learn to apply the method. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I intend to use this method in preference to the standard requirements analysis methods taught in RE course if I have to work on extracting the requirements from customer's problem in the future. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Submit

Appendix F - Student's post-task questionnaire on existing analysis method

I am a PhD student in the faculty of Computer Science and Information Technology, University of Malaya, Malaysia. This study aims to collect the student's perceptions about an existing analysis method. It is advisable that you learn and apply the method before answering the questionnaire. All the information acquired from this study will be used solely for academic purposes. Your participation and Co-operation in this study are greatly appreciated.

* Required

Which year you are studying in? *

- 1st
- 2nd
- 3rd

Do you have any experience of working on industrial projects? *

- Yes
- No

Please rate the following questions according to your perception about the analysis method used.

How do you perceive method?

I found the procedure for applying the method complex and difficult. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that this method would reduce the effort required to learn to analyse and structure customer's problem into set of requirements. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

The requirements extraction process using this method is difficult for students to understand. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Without the tool the method is difficult to be used. *

- Strongly agree
- Agree

- Neither agree nor disagree
- Disagree
- Strongly disagree

This method would make it easier for students to verify the extracted requirements. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that the method is easy to be understood. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I found the method to be useful in learning problem structuring and analysis process. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

By using this method, it is more difficult to analyse and structure the customers' problem. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found that it is difficult to apply the method for the given problem. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I would definitely not use this method to perform problem structuring and analysis process in requirements engineering. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I found the guidance provided to train the method clear and easy to understand *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method does not provide an effective solution to the problem of teaching problem structuring and analysis. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Overall, I think this method is an improvement to the standard requirements analysis methods taught in RE course. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree

- Strongly disagree

I am not confident that I am now competent to apply this method in practice. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

By using the software tool, it can be difficult to learn to apply the method *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

I intend to use this method in preference to the standard requirements analysis methods taught in RE course if I have to work on extracting the requirements from customer’s problem in the future. *

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

Submit